DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

SCHEME FOR
BACHELOR OF TECHNOLOGY (B.Tech)

COMPUTER SCIENCE & ENGINEERING
(Artificial Intelligence & Machine Learning)
(1st to 8th Semester)

Graduation Batch: 2024-2028

(With effect from 2024-25)

Dayananda Sagar University
APPROVED COPY

This was approved by
[illegible] 6th May 2026

SCHEME - B.TECH – 2024-25 ONWARDS

I SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	24EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	24EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2024-25 ONWARDS

II SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	24EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	24EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt., Bengaluru- 562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence & Machine Learning)

III SEM – CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SL. NO	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			Credit
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	
				L	T	P	J				
1	BSC	24AM2301	Probability & Statistics	3	0	0	0	60	40	100	03
2	IPCC	24AM2302	Data Structures	3	0	2	0	60	40	100	04
3	IPCC	24AM2303	Computer Networks	3	0	2	0	60	40	100	04
4	PCC	24AM2304	Discrete Mathematics and Graph Theory	3	0	0	0	60	40	100	03
5	PCC	24AM2305	Computer Organization and Architecture	3	0	0	0	60	40	100	03
6	PCC	24AM2306	Embedded Systems Design	3	0	0	0	60	40	100	03
7	SEC	24AM2307	Skill Enhancement Course – I (Java Programming)	1	0	2	0	100	--	100	02
8	PCC	24AM2308	AI for Renewable Energy Systems	1	0	0	0	100	--	100	01
9	AEC	24AM2309	Cognitive and Technical Skills-III	--	--	--	--	--	--	--	Pass/Fail
			Total	20	0	06	0				**23**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, BSC-Basic Science, L – Lecture, T – Tutorial, P – Practical, J –Project, C – No. Of Credits, AEC-Ability Enhancement Course

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt., Bengaluru- 562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence & Machine Learning)

V SEM – CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	IPCC	24AM3501	Machine Learning	3	0	2	0	60	40	100	04
2	IPCC	24AM3502	Operating Systems	3	0	2	0	60	40	100	04
3	PCC	24AM3503	Natural Language Models	3	1	0	0	60	40	100	04
4	PCC	24AM3504	Image Processing and Computer Vision	3	0	0	0	60	40	100	03
5	HSMC	24AM3505	Innovation and Entrepreneurship	2	0	0	0	60	40	100	02
6	SEC	24AM3506	Skill Enhancement Course –III (Cloud Computing)	1	0	2	0	100	--	100	02
7	PEC	24AM35XX	Professional Elective Course -I	2	0	0	2	100	--	100	03
8	AEC	24AM3507	Cognitive and Technical Skills-V	-	-	-	-	--	--	--	Pass/Fail
			Total	17	1	6	2				22

HSMC- Humanities and Social Sciences including Management Courses, IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, L – Lecture, T – Tutorial, P – Practical, J-Project, C – No. Of Credits, AEC-Ability Enhancement Course

AIML-SCHEME (2024-2028 (1ST - 8TH SEMESTER))

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt., Bengaluru- 562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence & Machine Learning)

VII SEM –CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	PROJ	24AM4701	Capstone Project Phase-I	0	0	0	8	60	40	100	04
2	PEC	24AM47XX	Professional Elective Course – IV	2	0	0	2	100	--	100	03
3	PEC	24AM47XX	Professional Elective Course – V/ MOOC	2	0	0	2	100	--	100	03
4	OEC	24OEXXXX	Open Elective – II	3	0	0	0	60	40	100	03
			Total	7	0	0	12				13

CID* - NPTEL COURSE ID

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, OEC-Open Elective Courses, PROJ-Project Work, L – Lecture, T – Tutorial, P – Practical, J–Project, C – No. Of Credits

Dayananda Sagar University

School of Engineering

Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt., Bengaluru- 562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Artificial Intelligence & Machine Learning)

PROFESSIONAL ELECTIVE COURSES DOMAIN-WISE

S. N	Domain-wise	Domain Clusters	PROFESSIONAL ELECTIVE COURSES									
			PEC-I		PEC-II		PEC-III		PEC-IV		PEC-V	
			5th Semester		6th Semester				7th Semester			
			Course Code	Course Name	Course Code	Course Name	Course Code	Course Name	Course Code	Course Name	Course Code	Course Name
1	Domain-1	Agentic AI	24AM3508	Ethics, Safety & Governance of Agentic AI	24AM3606	LLM based Agent Frameworks	24AM3611	Multi-Agent AI Systems	24AM4702	Autonomous Planning and Reasoning Agents	24AM4707	Operationalizing Agentic AI Systems
2	Domain-2	Vision, Speech and Multimodal Intelligence	24AM3509	Intelligent Speech Processing Systems	24AM3607	Visual Computing and Graphics	24AM3612	Intelligent Computational Imaging and Reconstruction	24AM4703	Intelligent Video Surveillance Systems	24AM4708	Multi-modal interaction and speech interfaces
3	Domain-3	Robotics and Automation	24AM3510	Foundations of Robotics and Intelligent Systems	24AM3608	Robot Operating Systems (ROS)	24AM3613	Autonomous Robotic Systems	24AM4704	Foundation Models for Robotics	24AM4709	Human-Centric Automation and Industry 5.0
4	Domain-4	Data Analytics and Applications	24AM3511	Explainable AI	24AM3609	AI for Autonomous Vehicles & Smart Mobility	24AM3614	AI for Healthcare	24AM4705	AI for Agriculture	24AM47010	Blockchain for Cybersecurity and Data Privacy
5	Domain-5	AI Language Intelligence	24AM3512	Cognitive Linguistics	24AM3610	Conversational AI and spoken Language Understanding	24AM3615	Building AI Applications with Large Language Models	24AM4706	Multi modal and Multi lingual Language Perception	24AM47011	Building Transformer-Based Natural Language Processing Applications
DLI INTEGRATED												
6	Domain-6	DLI/MOOC	Accelerated Computing		Deep Tech systems		Graphics and Simulation		AI and Data Science			

DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING



SYLLABUS FOR
BACHELOR OF TECHNOLOGY (B. Tech)

COMPUTER SCIENCE & ENGINEERING
(Artificial Intelligence & Machine Learning)
(3rd to 8th Semester)

Graduation Batch: 2024-2028

SCHEME - B.TECH – 2024-25 ONWARDS

I SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	24EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	24EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2024-25 ONWARDS

I SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1101	LINEAR ALGEBRA AND DIFFERENTIAL EQUATIONS	3	0	0	0	3	3	60	40	100
2	24EN1102	OBJECT ORIENTED PROGRAMMING	2	1	2	0	4	3	60	40	100
3	24EN1103	COGNITIVE AND TECHNICAL SKILLS – I	0	0	4	0	2	1	100	-	100
4	24EN1111	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	24EN1112	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	24EN1113	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	2	60	40	100
7	24EN1114	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	24EN1115	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation, SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2024-25 ONWARDS

II SEM - CHEMISTRY CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	24EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	24EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	24EN1104	ENGINEERING CHEMISTRY	2	0	2	0	3	3	60	40	100
5	24EN1105	INTRODUCTION TO MECHANICAL ENGINEERING	2	0	2	0	3	3	60	40	100
6	24EN1106	INTRODUCTION TO ELECTRICAL ENGINEERING	2	0	0	0	2	3	60	40	100
7	24EN1107	ENGINEERING MECHANICS	2	0	0	0	2	3	60	40	100
8	24EN1108	TECHNICAL ENGLISH	2	0	0	0	2	1	100	-	100
9	24EN1109	ENVIRONMENTAL SCIENCE	1	0	0	0	0	1	50	-	50
10	24EN1110	KANNADA KALI / MANASU	1	0	0	0	0	1	50	-	50
			17	1	10	0	21		660	240	900

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

SCHEME - B.TECH – 2024-25 ONWARDS

II SEM - PHYSICS CYCLE

SL	COURSE CODE	COURSE TITLE	SCHEME OF TEACHING					Duration in Hrs Exam	Examination (Maximum Marks)		
			L	T	P	J	C		CIE	SEE	TM
1	24EN1201	SINGLE AND MULTIVARIATE CALCULUS	3	0	0	0	3	3	60	40	100
2	24EN1202	C PROGRAMMING FOR PROBLEM SOLVING	2	1	2	0	4	3	60	40	100
3	24EN1203	COGNITIVE AND TECHNICAL SKILLS – II	0	0	4	0	2	1	100	-	100
4	24EN1111	ENGINEERING PHYSICS	3	0	2	0	4	3	60	40	100
5	24EN1112	INTRODUCTION TO ELECTRONICS ENGINEERING	3	0	0	0	3	3	60	40	100
6	24EN1113	ENGINEERING GRAPHICS AND DESIGN THINKING	2	0	2	0	3	2	60	40	100
7	24EN1114	BIOLOGY FOR ENGINEERS	2	0	0	0	2	1	100	-	100
8	24EN1115	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1	0	0	0	0	1	50	-	50
			16	1	10	0	21		550	200	750

L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. of Credits, CIE – Continuous Internal Evaluation,
SEE- Semester End Examinations, TM – Total Marks.

III SEM – CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

SL. NO	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	BSC	24AM2301	Probability & Statistics	3	0	0	0	60	40	100	03
2	IPCC	24AM2302	Data Structures	3	0	2	0	60	40	100	04
3	IPCC	24AM2303	Computer Networks	3	0	2	0	60	40	100	04
4	PCC	24AM2304	Discrete Mathematics and Graph Theory	3	0	0	0	60	40	100	03
5	PCC	24AM2305	Computer Organization and Architecture	3	0	0	0	60	40	100	03
6	PCC	24AM2306	Embedded Systems Design	3	0	0	0	60	40	100	03
7	SEC	24AM2307	Skill Enhancement Course – I (Java Programming)	1	0	2	0	100	--	100	02
8	PCC	24AM2308	AI for Renewable Energy Systems	1	0	0	0	100	--	100	01
9	AEC	24AM2309	Cognitive and Technical Skills-III	--	--	--	--	--	--	--	Pass/Fail
			Total	20	0	06	0				**23**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, BSC-Basic Science, L – Lecture, T – Tutorial, P – Practical, J –Project, C – No. Of Credits, AEC-Ability Enhancement Course

PROBABILITY AND STATISTICS

[As per Choice Based Credit System (CBCS)scheme]

SEMESTER – III

Course Code	:	24AM2301	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1. **Apply** statistical principles and probability concepts to solve complex problems in real-world scenarios involving uncertainty and randomness.
2. **Evaluate** and select appropriate probability distributions and statistical techniques toanalyze and interpret data accurately in various applications.
3. **Justify** the use of estimation methods and hypothesis testing techniques for drawing meaningful inferences about population parameters.
4. **Analyze** and interpret sample test results for different statistical relationships, such asmeans, variances, correlation coefficients, regression coefficients, goodness of fit, and independence, to make informed decisions.
5. **Identify** sample tests using appropriate statistical procedures to investigate the significance of observed data and communicate findings effectively.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in theclass.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students tocome up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I : Probability	**09 Hours**
Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem, Bayes' Theorem of Probability	

UNIT – II: Random Variables and their Properties and Probability Distributions	**09 Hours**
Discrete Random Variable, Continuous Random Variable, Joint Probability Distributions Their Properties, Probability Distributions: Discrete Distributions: Binomial, Poisson Distributions and their Properties; Continuous Distributions: Exponential ,Normal, Distributions and their Properties.	
UNIT – III: Estimation and testing of hypothesis	**06 Hours**
Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.	
UNIT – IV: Sample Tests-1	**07 Hours**
Large Sample Tests Based on Normal Distribution , Small Sample Tests : Testing Equality of Means, Testing Equality of Variances, Test of Correlation Coefficient	
UNIT – V: Sample Tests-2	**08 Hours**
Test for Regression Coefficient; Coefficient of Association, 2 – Test for Goodness of Fit, Test for Independence.	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** the principles of probability to solve complex problemsin various real-world scenarios.	L2 & L3
2	**Solve** and **compare** different probability distributions, including discrete and continuous random variables, in order to make informed decisions and predictions.	L2 & L3
3	**Apply** statistical estimation techniques, such as maximum likelihood estimation and interval estimation, to draw meaningful inferences about population parameters from sample data.	L3
4	**Examine** hypothesis testing methods, including large and small sample tests, to assess the significance of observed dataand draw valid conclusions.	L4
5	**Analyze** statistical relationships and perform sample tests to assess the Equality of means in different populations, Correlation coefficients between variables to determine the strength and direction of the relationship. Independence of variables using appropriate statistical tests to assess the absence of any relationship.	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2		2				1					
CO2	3	2	2		2				1					
CO3	3	2	2						1					
CO4	3	2	2		2				1					
CO5	3	2	2		2				1					

3: Substantial (High)2: Moderate (Medium) 1: Poor (Low)

TEXT BOOKS:

1. Probability & Statistics for Engineers and Scientists, Walpole, Myers, Myers, Ye. PearsonEducation.

REFERENCE BOOKS:

1. Probability, Statistics and Random Processes T. Veerarajan Tata McGraw – Hill
2. Probability & Statistics with Reliability, Queuing and Computer Applications, Kishor S. Trivedi,Prentice Hall of India ,1999

E-Resources:

1. https://nptel.ac.in/courses/106104233
2. https://nptel.ac.in/courses/117103067
3. https://nptel.ac.in/courses/103106120
4. https://www.coursera.org/learn/probability-intro#syllabus
5. https://nptel.ac.in/courses/111104073

Activity Based Learning (Suggested Activities in Class)

1. Tools like Python programming, R programming can be used which helps student to developa skill to analyze the problem and providing solution.
2. Regular Chapter wise assignments/ Activity/Case studies can help students to have criticalthinking, developing an expert mind set, problem-solving and teamwork.

Following are Assignments/ Activities Can be carried out using either R programming language orPython Programming or excel solver.

1. There are n people gathered in a room. What is the probability that at least 2 of them will havethe same birthday? (Use excel solver, R Programming, Python Programming)
   a. Use simulation to estimate this for various n., and Produce Simulation Graph.
   b. Find the smallest value of n for which the probability of a match is greater than 0.5.
   c. Explore how the number of trials in the simulation affects the variability of our estimates.
2. **Case Study 1: Customer Arrivals at a Coffee Shop**

a. A coffee shop wants to analyze the number of customer arrivals during its morning rush hour (7:00 AM to 9:00 AM). The shop has been recording the number of customer arrivals every 15 minutes for the past month.
b. Data: The data consists of the number of customer arrivals recorded at the coffee shop during each 15-minute interval for the past month.
c. Here is a sample of the data:

Time Interval	stomer Arrivals
00 AM - 7:15 AM	**6**
15 AM - 7:30 AM	**4**
30 AM - 7:45 AM	**9**
45 AM - 8:00 AM	**7**
00 AM - 8:15 AM	**5**
15 AM - 8:30 AM	**8**
30 AM - 8:45 AM	**10**
45 AM - 9:00 AM	**6**

analyze the customer arrivals and determine the probability distribution that best fits thedata. Specifically, explore both discrete and continuous probability distributions,

including the binomial, Poisson, exponential, and normal distributions.

3. **Case Study 2:** Comparing the Performance of Two Groups
   a. Suppose you are a data analyst working for a company that manufactures a new energy drink. The marketing team conducted a promotional campaign in two different cities (City A and City B) to determine the effectiveness of the campaign in increasing sales. The sales data for a random sample of customers in each city was collected over a week. Your task is to compare the average sales between the two cities and test whether thereis a significant difference in the variance of sales.
   b. **Data**: Let's assume the following sample data for the number of energy drinks sold in each city:

City A: [30, 28, 32, 29, 31, 33, 34, 28, 30, 32]
City B: [25, 24, 26, 23, 22, 27, 29, 30, 26, 24]

perform a two-sample t-test to test the equality of means and a test for equality of variances using Python's SciPy library.

4. **Case study 3:** testing independence between two categorical variables.
   a. Data: Sample of 100 employees, and each employee is classified as either Male or Female.They were asked to rate their job satisfaction on a scale of 1 to 5, where 1 represents low satisfaction and 5 represents high satisfaction. The data is as follows:

Employee	Gender	Job Satisfaction
1	Male	4
2	Female	3
3	Male	2
4	Female	5
...	...	...

100	Female	4

b. Test for independence between gender and job satisfaction, use the chi-squared test in R.

****

DATA STRUCTURES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	:	24AM2302	**Credits**	:	04
Hours / Week	:	05 Hours	**Total Hours**	:	39(Th)+26(P) Hours
L–T–P-J	:	3–0–2-0			

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

This Course will enable students to:

1. **Understand** the basic approaches for analyzing and designing data structures.
2. **Introduce** dynamic memory allocation and C language concepts required for building data structures.
3. **Develop** essential skills to construct data structures to store and retrieve data quickly and **efficiently**.
4. **Utilize** different data structures that support different sets of operations which are suitable for various applications.
5. **Explore & implement** how to insert, delete, search, and modify data in any data structure- Stack, Queues, Lists, Trees.
6. **Develop** applications using the available data structure as part of the course for mini project.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**
IINTRODUCTION: Introduction to Data Structure, Classification, C Structure and Union, C Pointers, Array Definition, Representation, Operations (Insertion, Deletion, Search and Traversal), Two/Multidimensional Arrays, Sparse matrix. TB1: 1.1, 2.2, 2.5 ; TB2: 1.1, 1.2,1.3.1-1.3.4; RB1: 5.1 – 5.12, 6.4	
UNIT – II	**08 Hours**
INTRODUCTION TO ADT: **Stack:** Definition, Array Representation of Stack, Operations on Stacks. **Applications of Stack:** Expression evaluation, Conversion of Infix to Postfix, Infix to Prefix Recursion, Tower of Hanoi **Queue:** Definition, Representation of Queues, Operations of Queues, Circular Queue. **Applications of Queue:** Job Scheduling, A Maze Problem TB1: 3.1,3.2, 3.3,3.4,3.5 ; TB2: 2.1, 2.2, 2.3, 3.2, 3.3	
UNIT – III	**08 Hours**
DYNAMIC DATA STRUCTURES: **Linked List:** Types, Representation of Linked Lists in Memory. Traversing, Searching, Insertion & Deletion from Linked List. Circular List, Doubly Linked List, Operations on Doubly Linked List (Insertion, Deletion, Traversal). **Applications:** Stack & Queue Implementation using Linked Lists. **Case Study**: Josephus problem. TB2: 4.2,4.3,4.5	
UNIT – IV	**08 Hours**
TREES: Basic Terminology, Binary Trees and their representation, Complete Binary Trees, Binary Search Trees, Threaded Binary Trees, Operations on Binary Trees (Insertion, Deletion, Search & Traversal). **Applications:** Expression Evaluation **Case Study:** Game Tree TB2: 5.5.3,5.5.4,5.6	
UNIT – V	**07 Hours**
Efficient Binary Search Trees: Optimal Binary Search Trees, AVL Trees, Red Black Trees, Splay Trees. **Case Study**: B Trees TB1: 10.1,10.2,10.3,10.4, 11.2	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Apply the concepts of pointers, arrays, structures, and unions to address real-world problems and implement the concept in C programming language.	L3
2	Utilize stacks and queue data structures to solve problems such as infix to postfix, infix to prefix conversions, the Towers of Hanoi puzzle, job scheduling and maze navigation.	L3
3	Implement and manipulate singly linked lists, doubly linked lists, and circular linked lists, executing operations such as insertion, deletion, and traversal.	L3
4	Understand the concepts of binary trees, binary search trees, and threaded binary trees, and their associated operations.	L2
5	Understand advanced binary tree structures includes optimal binary search trees, AVL trees, Red-Black trees, and Splay trees.	L2

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO2	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO3	1	2	3	-	3	-	-	-	-	-	-	-	2	2
CO4	-	1	2	-	3	-	-	-	-	-	-	-	2	2
CO5	-	1	2	-	-	-	-	-	-	-	-	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

1. Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, "Fundamentals of Data structures in C", 2nd Edition, Orient Longman, 2008.
2. A.M. Tannenbaum, Y Langsam, M J Augentien "Data Structures using C", 1st Edition, Pearson, 2019.

REFERENCE BOOKS:

1. Brian. W. Kernighan, Dennis. M. Ritchie, "The C Programming Language", 2nd Edition, Prentice-Hall, 1988.
2. Gilbert & Forouzan, "Data Structures: A Pseudo-code approach with C", 2nd Edition, Cengage Learning,2014.
3. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, McGraw Hill, 2013.
4. R.L. Kruse, B.P. Learly, C.L. Tondo, "Data Structure and Program design in C", 5th Edition, PHI ,2009.

E-Resources:

1. https://nptel.ac.in/courses/106102064
2. https://www.coursera.org/learn/data-structures?specialization=data-structures-algorithms
3. https://www.udemy.com/topic/data-structures/free/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/data-structures
5. https://cse01-iiith.vlabs.ac.in/
6. https://kremlin.cc/k&r.pdf

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.
4. Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

DATA STRUCTURES LABORATORY

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language.

1. To Implement C programs to perform array operations.
2. To determine the validity of a 9x9 Sudoku board (application of 2-dimensional array).
3. To store, retrieve and update the elements in structures (structures and pointers to structures).
4. To implement stack using linked list.
5. To implement a queue data structure using a singly linked list.
6. To implement a singly linked list and its operations.
7. To implement a doubly linked list and its operations.
8. To create a circular queue using a circular linked list data structure
9. To implement binary tree traversal techniques.

OPEN-ENDED EXPERIMENTS

1. Design a web browser history tracker in C. Implement a stack data structure to keep track of visited URLs. Create functions to push new URLs onto the stack as users visit websites and pop URLs when users navigate backward in their browsing history.

2. Imagine you are responsible for designing a queue-based system to manage the queue of regular customers waiting to purchase cinema tickets at a popular movie theatre. Your system should ensure fair and efficient ticket sales for all customers. When a customer's arrive at the cinema, they join the queue. Each customer is represented by his name, age (for record-keeping), and number of tickets needed. When a customer reaches the front of the queue, they are served by the ticketing agent. Implement a ticket sale process where the agent provides the customer with the requested ticket(s). Initialize the total number of tickets and if the tickets are sold, then the ticketing agent should display a houseful message.

COMPUTER NETWORKS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER –III

Course Code	: 24AM2303	**Credits**:	04
Hours / Week	: 05 Hours	**Total Hours**:	39(T)+26(P) Hours
L–T–P-J	: 3–0–2-0		

Course Learning Objectives:

This Course will enable students to:

1. **Outline** the basic principles of computer networking and how computer network hardware and software operate.
2. **Evaluate** the operation and performance of practical data link protocols using the principles of framing, error detection and correction.
3. **Apply** the principles of network layer design to the analysis and evaluation of routing algorithms, congestion control techniques, internetworking and addressing.
4. **Investigate** the basic transport layer facilities and essentials of transport. protocol
5. **Illustrate** the working of various application layer protocols.

Teaching-Learning Process (General Instructions)

1. These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.
2. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
3. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
4. Show ***Video/Animation*** films to explain functioning of various concepts.
5. Encourage ***Collaborative*** (Group Learning) Learning in the class.
6. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
7. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
8. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
9. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION: Networks, network types, internet history, standards and administration (TB1-Ch1); Network models: Protocol layering, TCP/IP protocol suite, the OSI model (TB1-Ch2); Transmission media: Introduction, guided media, unguided media (TB1-Ch7); Switching: Introduction, circuit-switched networks, packet switching (TB1-Ch8).
UNIT – II **08 Hours**
Link layer addressing; (TB1-Ch10) Error detection and correction: Cyclic codes, checksum, forward error correction; (TB1-Ch10) Data link control: DLC services, data link layer protocols; (TB1-Ch11 & TB2-Ch3) Media access control: Random access, virtual LAN. (TB1-Ch12, Ch15)
UNIT – III **08 Hours**
Network layer design issues; (TB2-Ch5) Routing algorithms (Distance Vector Routing, Link State Routing and Hierarchical Routing).; (TB2-Ch5) Congestion control algorithms; (TB2-Ch5) Quality of service, and internetworking; (TB2-Ch5) The network layer in the internet: IPv4 addresses, IPv6; (TB2-Ch5, TB1-Ch19) Internet control protocols, OSPF (Open Shortest Path First), IP (Internet Protocol); (TB2-Ch5)
UNIT – IV **08 Hours**
The transport service, elements of transport protocols; (TB2-Ch6) Congestion control; (TB2-Ch6) The internet transport protocols: UDP (User Datagram Protocol), TCP (Transport Control Protocol); (TB2-Ch6) Performance problems in computer networks, and network performance measurement. (TB2-Ch6)
UNIT – V **07 Hours**
Introduction, client server programming, WWW (World Wide Web) and HTTP (Hyper Text Transfer Protocol); (TB1-Ch27) FTP (File Transfer Protocol); (TB1-Ch26) E-mail, telnet, (TB1-Ch26 & TB2-Ch7) DNS (Domain Naming System); (TB2-Ch7) SNMP (Simple Network Management Protocol) (TB1-Ch28)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Explain** the basic concepts of data communications including the key aspects of networking and their interrelationship, packet switching, circuit switching and cell switching as internal and external operations, physical structures, types, models, and internetworking.	L2

2	**Apply** the concept of Hamming distance, the significance of the minimum Hamming distance and its relationship to errors as well as the detection and correction of errors in block codes.	L3
3	**Solve** the problems related to various Routing Algorithms and also perform the Interpretation of routers, Internet Protocol IPv4, and IPv6.	L3
4	**Recognize** transport layer services and infer UDP and TCP protocols and **Distinguish** between UDP and TCP Protocols.	L4
5	**Infer** the significance, and purpose of protocols (FTP, SMTP), standards, and use in data communications and networking and **analyze** the most common DNS resource records that occur in a zone file.	L4

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	-	-	-	-	-	-	-	-	-	-	-	1	1
CO2	3	3	3		-	-	-	-	-	-	-	-	2	2
CO3	3	3	3	-	-	-	-	-	-	-	-	-	2	2
CO4	3	3	3	-	-	-	-	-	-	-	-	-	1	1
CO5	3	3	3	-	-	-	-	-	-	-	-	-	1	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Behrouz A. Forouzan—Data Communications and Networking with TCP/IP Protocol Suite‖ TataMcGraw- Hill, 6th Edition, 2022.

2. Andrew S. Tanenbaum, David Wetherall, —Computer Networks|| Prentice-Hall, 6th Global Edition, 2021.

REFERENCE BOOKS:

1. Chwan-Hwa Wu, Irwin, —Introduction to Computer Networks and Cyber Security||, CRC publications, 2014.
2. Douglas E. Comer, —Internetworking with TCP/IP —, Prentice-Hall, 5thEdition,2011.
3. Peterson, Davie, Elsevier, —ComputerNetworks,5thEdition,2011
4. Comer, —Computer Networks and Internets with Internet Applications,4thEdition,2004.

E-Resources:

1. http://computer.howstuffworks.com/computer-networking-channel.htm
2. https://www.geeksforgeeks.org/layers-osi-model/
3. https://www.wikilectures.eu/w/Computer_Network
4. https://technet.microsoft.com/en-us/network/default.aspx

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Flip class activity

LABORATORY EXPERIMENTS

1. Analyse the various line coding techniques used for data transmission of a digital signal over a transmission line
2. Design a program for error-detecting code using CRC-CCITT (16- bits).
3. Design a program to find the shortest path between vertices using Belman- ford algorithm
4. Given a graph derive the routing table using distance vector routing and link state routing algorithm
5. Try out some simple subnetting problems.
6. Using TCP/IP sockets, write a client–server program to make the client send the file name and to make the server send back the contents of the requested file if present. Implement the above program using message queues or FIFOs as IPC channels

7. Implement a webserver program to fetch a URL request and display the home page of the same in the browser
8. Implement a simple DNS server to resolve the IP address for the given domain name

DISCRETE MATHEMATICS AND GRAPH THEORY
[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Subject Code	:	24AM2304	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–J	:	3–0–0–0			

Course Learning Objectives:

This Course will enable students to:

1**. Learn** the set theoretic concept and its application in theory of computation.

2. **Determine** the concepts of mathematical induction, recursive relations and their application.

3. **Illustrate** the association of functions, relations, partial ordered set and lattices with problems related to theoretical computer science and network models.

4. **Discuss** the basics of graph theory and its application in computer networks. Learn the concepts of counting techniques and its application.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. **Lecture method** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** Adopt the Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, ask at least three Higher order Thinking questions in the class.
6. Adopt **Problem Based Learning**, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every **concept can be applied to the real world** - and when that possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

SET THEORY: Sets and subsets, Operations on Sets: Basic set operations, algebraic properties of sets, The Addition Principle

RELATIONS AND ITS PROPERTIES: Relations and their properties, N-Ary Relations

and their applications, Representing relations. ***Textbook – 2: 1.1, 1.2 ; Textbook – 1: 7.1., 7.2, 7.3***
UNIT – II **06 Hours**
RELATIONS AND ORDER RELATIONS: Closure of relations, Equivalence Relations, Partial Orderings, Functions, The Growth of Functions. **Self-Study:** Transitive Closure and Warshall's Algorithm. ***Textbook – 1: 7.4., 7.5, 7.6, 3.2***
UNIT – III **08 Hours**
MATHEMATICAL INDUCTION AND RECURSION: Mathematical Induction, Recurrence Relations: Rabbits and the Fibonacci Numbers, The Tower of Hanoi, Code word Enumeration, Solving Linear Recurrence Relations **Self-Study:** Basic Connectives and Truth Tables ***Textbook-1: 4.1;6.1, 6.2;1.1***
UNIT – IV **09 Hours**
GRAPH THEORY: Graphs and Graph Models. Graph Terminology and Special Types of Graphs: Basic Terminology, Some Special Simple Graphs, Bipartite Graphs, Complete Bipartite Graphs. Representing Graphs and graph isomorphism: Adjacency lists, Adjacency Matrices, Incidence Matrices, Connectivity: Paths, Connectedness in Undirected and Directed Graphs, Vertex and Edge connectivity and their applications. ***Textbook-1: 8.1, 8.2, 8.3, 8.4***
UNIT – V **08 Hours**
GRAPHS AND ITS APPLICATIONS: Euler and Hamilton Paths and their applications, Planar Graphs and their Applications, Graph Coloring and its applications. ***Textbook-1: 8.5, 8.7, 8.8***

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Identify** the membership of the Set, Relations and make use of basic Algebraic properties.	L3
2	**Examine** the steps involved in Mathematical Induction and Linear recurrence-related problems.	L4
3	**Construct** different types of graphs based on the properties and the real-time applications of graph theoretical concepts.	L3
4	**Analyze** the methods for optimizing the solution for graph coloring problem, Eulerian and Hamilton circuits/planes.	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	1	2					1	1	1		2	1	0
CO2	3	3	2					1	1	1		2	1	0
CO3	3	3	3					1	1	1		1	1	0
CO4	3	3	3					1	1	1		1	1	0

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Kenneth H. Rosen, "Discrete Mathematics and its applications", Tata McGraw Hill, 2003.
2. Bernard Kolman, Robert C. Busby, Sharon Ross, "Discrete Mathematical Structures", 3rdEdition, PHI 2001.

REFERENCE BOOKS:

1. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", IV Edition, Pearson Education, Asia, 2002.
2. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with applications to computer Science", Tata McGraw Hill, 1987.
3. J K Sharma, "Discrete Mathematics", 3rd edition, 2013, Macmillan India Ltd.

E-Resources:

1. Discrete Mathematics with Algorithms by M. O. Albertson, J. P. Hutchinson – J 1988, Wiley.
2. Discrete Mathematics for Computer Science, Gary Haggard, John Schlipf, Sue Whitesides,Thomson Brooks/Cole, 2006.
3. http://ocw.mit.edu/courses/mathematics/
4. http://www.nptelvideos.in/2012/11/discrete-mathematical-structures.html
5. http://cglab.ca/~discmath/notes.html
6. https://www.cs.odu.edu/~toida/nerzic/content/web_course.html

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion.
2. Demonstration of solution to a problem using graph theory.

****

COMPUTER ORGANIZATION AND ARCHITECTURE
[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 24AM2305	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P-J	: 3–0–0-0		

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the Architecture and programming of ARM microprocessor.
2. **Develop** program using Arm instruction set and appreciate the advanced features provided in the ARM
3. **Understand** the exception handling techniques.
4. **Study in** detail the concept of instruction level parallelism and concepts of pipelining.
5. **Understand** various cache memory mapping techniques and memory Organization.

Teaching-Learning Process

1. ***Lecture method*** along with traditional lecture method, different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *incorporating* brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.

UNIT – I	**05 Hours**

An Overview of Computing Systems: History of Computers, The Computing Device,
The ARM7TDMI Programmers' Model:
Introduction, Data types, Processor Modes, Registers, Program Status Registers, The vector Table.
Assembler Rules and Directives: Structure of Assembly Language Modules, Registers, Directives and Macros. **Loads, Stores and Addressing:** LODS and STORES instructions, Operand Addressing , ENDIANNES ***Text Book-1: 1.1 to 1.3; 2.1 to 2.6 ; 4; 5.3, 5.4, 5.5***

UNIT – II	**05 Hours**

Constants and Literal Pools: The ARM Rotation Scheme, Loading Constants and address into Registers **Logic and Arithmetic:** Flags and their Use, Compare instructions, Data Processing Instructions **Loops and Branches:** Branching, Looping, Conditional Execution, Straight-Line Coding **Subroutines and Stacks:** Stack, Subroutines, Passing parameters to subroutines, The ARM APCS. ***(Text Book-1: 6.1 to 6.4; 7.1 to 7.4; 8.2 to 8.6; 10.1 to 10.5***

UNIT – III	**05 Hours**
Mixing C and Assembly Language: Inline Assembler Embedded Assembler, Calling Between C and Assembly. **Exception Handling:** Interrupts, Error Conditions, Processor Exception Sequence, The Vector Table, Exception Handlers, Exception Priorities, Procedures for Handling Exceptions. ***(Text Book-1: 11.1 to 11.8; 14.1 to 14.4***	

UNIT – IV	**12 Hours**
Pipelining: Basic and Intermediate Concepts Introduction, The Major Hurdle of Pipelining, How Pipelining Implemented, What makes Pipelining hard to Implement, Extending the MIPS Pipeline to Handle Multicycle Operations, The MIPS R4000 Pipeline, Crosscutting Issues. ***Text Book-2: C.1 to C.7***	

UNIT – V	**12 Hours**
Memory Hierarchy: *Introduction, Cache Performance, Six basic cache Optimizations, Virtual Memory, Protection and examples of Virtual Memory, Fallacies and Pitfalls.* ***Text Book-2: B.1 to B.6***	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	**Apply** knowledge of the internal architecture and organization of ARM microprocessors to utilize their components and functionalities.	L3
2	**Apply** the instruction set of ARM Microprocessor by writing Assembly language programs.	L3
3	**Analyze and compare** the various exception handling techniques.	L4
4	**Examine** the concept of instruction-level parallelism and **analyze** the principles of Pipelining techniques.	L4

5	**Compare** and C**ontrast** memory hierarchy and its impact on computer cost/performance.	L4

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3		2										2	
CO2	3		3		1								2	
CO3	3	3	1										2	
CO4	3	3	1										2	
CO5	3	3	1										2	

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. William Hohl, "ARM Assembly Language", 2nd Edition, CRC Press, 2009.
2. John L Hennessy, David A Patterson, "Computer Architecture, A Quantitative Approach", 5th Edition, Morgan Kaufmann publishers, 2012.

REFERENCE BOOKS:

1. David A Patterson, John L Hennessy, "Computer Organization and Design", 4th Edition, Morgan Kaufmann publishers, 2010.
2. Steve Furber, "ARM System-on-chip Architecture", 2nd Edition, Pearson Publications, 2000.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", 5th Edition, Tata McGraw Hill, 2002.

E-Resources:

1. https://www.udemy.com/topic/arm-cortex-m/
2. https://www.edx.org/school/armeducation
3. https://onlinecourses.nptel.ac.in/noc22_cs93/preview

Activity Based Learning (Suggested Activities in Class)

1. Mini project implementation using Assembly Language Programming.
2. Demonstration of solution to a problem through programming.

EMBEDDED SYSTEMS DESIGN

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 24AM2306	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P-J	: 3–0–0-0		

Course Learning Objectives:

This Course will enable students to:

1. **Understand** the fundamental concepts of embedded system design.
2. Gain **knowledge** of various hardware and software components used in embedded systems.
3. **Develop** skills to design and implement embedded systems for different applications.
4. Learn to **analyze** and optimize the performance of embedded systems.
5. **Enhance** problem-solving and critical thinking abilities in the context of embedded system design.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Interactive Lectures:*** Engage students through discussions, case studies, and real-life examples.
2. ***Hands-on Projects:*** Assign practical projects to students to enhance their understanding and application of concepts.
3. ***Group Discussions:*** Encourage collaborative learning and problem-solving through group discussions and brainstorming sessions.
4. ***Case Studies:*** Analyze real-world embedded system designs to understand their challenges and solutions.
5. ***Simulations and Virtual Labs:*** Use simulation tools and virtual labs to provide a virtual hands-on experience.
6. ***Guest Lectures:*** Invite industry experts to share their experiences and provide insights into real-world embedded system design practices.
7. ***Online Forums:*** Establish an online platform for students to discuss and share their ideas and questions related to the course.
8. ***Demonstrations:*** Conduct live demonstrations of embedded system prototypes to showcase practical implementations.
9. ***Assignments and Assessments:*** Assign regular assignments and assessments to evaluate students' understanding and progress.
10. ***Industry Visits:*** Organize visits to embedded system manufacturing companies to expose students to real-world applications.

UNIT – I	**07 Hours**
INTRODUCTION TO EMBEDDED SYSTEMS **Introduction:** What is an Embedded System, Embedded Systems VS. General Computing Systems, History of Embedded Systems Classification of Embedded Systems, Major Application Areas of Embedded Systems, Purpose of Embedded Systems, Wearable Devices—The Innovative Bonding of Lifestyle with Embedded Technologies ***(Text Book-3: Chapter 1)*** **Characteristics and Quality Attributes of Embedded Systems:** Characteristics of an Embedded System, Quality Attributes of Embedded Systems ***(Text Book-3: Chapter 3)*** **Embedded Systems—Application- and Domain-Specific:** Washing Machine—Application-Specific Embedded System, Automotive–Domain Specific Examples of Embedded System ***(Text Book-3: Chapter 4)***	
UNIT – II	**08 Hours**
EMBEDDED SYSTEM HARDWARE DESIGN-I **Embedded System Core:** General Purpose and Domain Specific Processors, Application Specific c Integrated Circuits (ASICs), Programmable Logic Devices (PLDs), Commercial off-the-shelf Components (COTS) ***(Text Book 3: Chapter 2.1)*** **Memory:** Overview on Various Types of memory sub systems used in Embedded systems and their selection ***(Text Book 3: Chapter 2.2)*** **Sensors and Actuators:** interfacing of LEDs, 7-segment LED Displays, Piezo Buzzer, Stepper Motor, Relays, Optocouplers, Matrix keyboard, Push button switches, Programmable Peripheral Interface Device (e.g. 8255 PPI), etc. with the I/O subsystem of the embedded system. ***(Text Book 3: Chapter 2.3)***	
UNIT – III	**08 Hours**
EMBEDDED SYSTEM HARDWARE DESIGN-II **Communication Interface:** I2C, SPI, CAN, UART,1-wire, parallel bus, etc. RS-232C, RS-485, Parallel Port, USB, IEEE 1394, Infrared (IrDA), Bluetooth, Wi-Fi, ZigBee, GPRS, etc. ***(Text Book 3: Chapter 2.4)*** **Other System Components:** Reset Circuit, Brown-out protection circuit, Oscillator Unit, Real-Time Clock (RTC), Analog to Digital Converter (ADC), Timers and Watchdog Timer unit ***(Text Book 3: Chapter 2.6)*** **Arm Cortex Mx Processor Family Overview:** Features, Architecture, Memory System, Exception and Interrupts, Low Power Features ***(Text Book 1: Chapter 3)***	
UNIT – IV	**08 Hours**
EMBEDDED SYSTEM SOFTWARE DESIGN **Programming Concepts and Embedded Programming in C:** High -Level Language C programming, C program elements (compiler build stages, macros, functions, Bitwise Operations, Looping constructs, Pointers, and AAPCS) ***(Reference Book 2: Chapter 5.1 to 5.6)***	
UNIT – V	**08 Hours**

EMBEDDED SYSTEM SOFTWARE DESIGN
Embedded Firmware Design and Development: Embedded Firmware Design Approaches ***(Text Book 3: Chapter 9.1)***

REAL-TIME OPERATING SYSTEMS
Operating System Basics: The Kernel, Types of Operating Systems, Tasks, Process and Threads ***(Text Book 3: Chapter 10.1, 10.2, 10.3)***

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Understanding** the fundamentals of embedded systems, including their characteristics, classifications, application domains, and quality attributes.	L2
2	**Describe** the roles of hardware components in embedded systems.	L2
3	**Illustrate** the working of embedded communication interfaces, system support components, and basic architecture of ARM Cortex Mx processors.	L2
4	**Apply** embedded C programming concepts to develop software modules using functions, pointers, bitwise operations, and looping constructs in accordance with AAPCS standards.	L3
5	**Analyze** embedded firmware design approaches and the role of real-time operating system components such as kernels, tasks, processes, and threads in embedded applications.	L4

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
COs	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2											3	2
CO2	3	2			2								3	2
CO3		2	2		2								3	2
CO4	3	2	3	2	3								2	3

CO5	3	3	2	3	2	1							2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Joseph Yiu," The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors", 3rd Edition, Newnes, 2013
2. K.V. Shibu," Introduction to Embedded Systems", 2nd Edition, McGraw Hill Education, 2017.

REFERENCE BOOKS:

1. James K. Peckol, "Embedded Systems: A Contemporary Design Tool", Wiley, 2009.
2. Raj Kamal, "Embedded Systems- Architecture, Programming and Design", 3rd Edition, McGraw Hill Education, 2017.
3. Jonathan Valvano," Embedded Systems: Real-Time Operating Systems for ARM Cortex-M Microcontrollers", 2nd Edition, CreateSpace Independent Pub, 2012.

E-Resources:

1. MOOC Course: "Introduction to Embedded Systems" by University of California, Irvine (Link: [www.coursera.org/embedded- systems])
2. Website: Embedded.com (Link: [www.embedded.com])
3. Online Tutorial: "Embedded Systems Tutorial" by Tutorials point (Link: [www.tutorialspoint.com/embedded_system])
4. ARM Procedure Call Standard (AAPCS) Standard documentation (Link: https://developer.arm.com/documentation/dui0041/c/ARM-Procedure-Call-Standard)

Activity Based Learning (Suggested Activities in Class)

1. **Project-based Learning:** Assign a semester-long project where students design and implement an embedded system for a specific application.
2. **Hackathons:** Organize hackathons where students work in teams to solve a given problem using embedded system design techniques.
3. **Guest Speaker Series:** Invite professionals from the industry to share their experiences and projects related to embedded system design.

Hands-on:

1. Introduction to Embedded System & STM32FXX Kit.
2. Explain the step-by-step procedure to create, build, and upload a program to the STM32FXX microcontroller using Keil uVision5.
3. Developing the GPIO Output driver to toggle the Onboard LED (Using Inbuilt Headers).
4. Developing the GPIO Input driver to turn ON the Onboard LED when you press the onboard button.
5. Developing the ADC Driver Code.

SKILL ENHANCEMENT COURSE-I
JAVA PROGRAMMING
[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	: 24AM2307	**Credits**	: 02
Hours / Week	: 03 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 1–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. Understand the basic principles of object-oriented programming and their application in Java.
2. Demonstrate proficiency in writing and executing simple Java programs using fundamental programming constructs.
3. Apply problem-solving skills to design and implement Java programs that solve real-world computational problems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Use* ***real-world examples*** *and scenarios to demonstrate the practical relevance of Java programming concepts which enhances their understanding of how Java is used in real-world applications*
2. ***Interactive Coding Sessions:*** *Conduct* ***Interactive coding sessions*** *where students can code alongside the teacher or participate in coding challenges. This promotes active participation and helps students develop their coding skills*
3. ***Project-Based Learning****: Assign projects or mini-projects that require students to apply Java programming concepts and develop complete applications. This approach fosters independent learning, problem solving skills and a deeper understanding of Java Programming Principles.*
4. ***Active Learning Strategies:*** *Incorporate* ***active learning strategies*** *such as group discussions, problem-solving activities, case studies, and hands-on coding exercises. This allows students to actively engage with the material and apply their knowledge in practical scenarios*
5. ***Collaborative Learning:*** *Encourage students to work in pairs or small groups on programming tasks.* ***Collaborative learning*** *promotes peer-to-peer learning, fosters teamwork, and allows for the exchange of ideas and knowledge.*
6. ***Guest Speakers and Industry Connections:*** *Invite* ***guest speakers, industry professionals, or alumni*** *who work with Java programming to share their*

experiences and insights. This provides students with real-world perspectives and helps them understand the practical applications of Java programming	
UNIT – I	**06 Hours**
An Overview of Java: Object-Oriented Programming, A First Simple Program, A Second Short Program, Two Control Statements, Using Blocks of Code, Lexical Issues, The Java Class Libraries, **Data Types, Variables, and Arrays**: Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, A Closer Look at Literals, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays ***(Text book-1: Ch 2, Ch 3).***	
UNIT – II	**05 Hours**
Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses, **Control Statements**: Java's Selection Statements, Iteration Statements, Jump Statements. ***(Text book 1: Ch 4, Ch 5).***	
UNIT – III	**05 Hours**
Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize( ) Method, A Stack Class, **A Closer Look at Methods and Classes**: Overloading	
Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static ***(Text book 1: Ch 6, Ch 7).***	
UNIT – IV	**05 Hours**
Inheritance: Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class. Packages: **Packages**, Packages and member Access, Importing Packages, **Interfaces**, ***(Text book 1: Ch 8,Ch 9).***	
UNIT – V	**05 Hours**
Exception Handling: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Three Recently Added Exception Features, Using Exceptions. ***(Text book 1: Ch 10).***	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Demonstrate an understanding of object-oriented programming concepts in Java.	L2

2	Apply different operators and control statements in Java to solve programming problems.	L3
3	Design and implement classes, methods, and constructors in Java to create reusable and modular code.	L4
4	Utilize inheritance, packages, and interfaces in Java to develop complex object-oriented programs.	L4
5	Handle exceptions effectively in Java programs and develop robust error-handling mechanisms.	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2				1				2			3	
CO2		3	2			2				2				3
CO3	2	2	3						2	2		2		2
CO4	2	2	3			1							2	
CO5		2		2	2	2								2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Herbert Schildt, Java The Complete Reference, 11th Edition, Tata McGraw Hill, 2019. (Chapters 2, 3, 4, 5, 6,7, 8, 9,10)

REFERENCE BOOKS:

1. H.M.Dietel and P.J.Dietel,Java, How to Program, 11th Edition, Pearson Education/PHI, 2017

2. E Balagurusamy, Programming with Java A primer, McGraw Hill Education,2019.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs47
2. https://www.geeksforgeeks.org/java/
3. https://www.codecademy.com/learn/learn-java

Activity Based Learning (Suggested Activities in Class)

1. **Code Walkthrough and Debugging**:
   a) Provide students with a pre-written Java program that contains bugs or logical errors. Divide the class into small groups or pairs.
   b) Instruct each group to identify and fix the errors in the code, encouraging them to use their knowledge of Java programming concepts.
   c) After a set time, have each group present their fixed code to the class and explain their debugging process.
2. **Mini Java Project:**
   a) Divide the class into small groups and assign each group a specific programming problem or scenario that requires a Java solution.
   b) Instruct each group to design and implement a Java program to solve the assigned problem, utilizing concepts learned in class.
   c) Provide guidance and support as needed while encouraging students to apply their creativity and problem-solving skills.

****

AI FOR RENEWABLE ENERGY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – III

Course Code	:	24AM2308	**Credits**	:	01
Hours / Week	:	01 Hours	**Total Hours**	:	13 Hours
L–T–P-J	:	1–0–0-0			

Course Objectives:

This Course will enable students to:

1. **Understand** the fundamentals of renewable energy systems and their challenges.
2. **Apply** ML/DL models for energy forecasting and optimization.
3. **Design** AI-driven control systems for smart energy management.
4. **Develop** predictive maintenance models for renewable infrastructure.
5. **Integrate** AI with IoT for real-time monitoring of renewable systems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

9. ***Lecture method*** means it includes not only traditional lecture method, but different type of teaching methods may be adopted to develop the course outcomes.
10. ***Interactive Teaching***: Adopt the **Active learning** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
11. Show ***Video/animation*** films to explain functioning of various concepts.
12. Encourage ***Collaborative*** (Group Learning) Learning in the class.
13. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
14. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
15. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
16. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**03 Hours**

INTRODUCTION TO RENEWABLE ENERGY:

Overview of Renewable Energy Sources (Solar, Wind, Hydro, Biomass), Challenges in Renewable Energy Integration, Case Studies of Renewable Energy.

(Text Book-1 :)

UNIT – II	**03 Hours**

MACHINE LEARNING FOR ENERGY FORECASTING:

Time Series Forecasting Techniques (ARIMA), Supervised Learning for Energy Load Prediction, Feature Engineering with Weather & Environmental Data, Evaluation Metrics

(MAE, RMSE, R^2) ***(Text Book-2:)***
UNIT – III **02 Hours**
DEEP LEARNING IN RENEWABLE SYSTEMS: LSTM, GRU, and Hybrid Models for Energy Forecasting, CNN for Solar Panel Image Analysis, Predictive Maintenance with Sensor Data, Fault Detection and Classification in Solar/Wind Farms. ***(Text Book-3:)***
UNIT – IV **02 Hours**
Optimized Use of PV Arrays: Introduction to Optimization Algorithms, Efficiency of a MPPT Algorithm, Comparison of Different Algorithms ***(Ref boook-1:)***
UNIT – V **03 Hours**
Modelling of Energy Storage Systems and Real-Time Control: Description of Different Storage Systems, Battery Bank Systems, Battery Bank Model, Equivalent Circuit Battery Models. ***(Text Book-1 :)***

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Explain** the fundamentals of renewable energy systems such as solar, wind, hydro, biomass.	L2
2	**Apply** machine learning models to perform energy generation and load forecasting using historical and weather-related data.	L3
3	**Analyze** the outputs of machine learning/deep learning models for fault detection and predictive maintenance in renewable energy systems using sensor or image data.	L4
4	**Compare** commonly used optimization and MPPT algorithms to assess their impact on the efficiency of photovoltaic (PV) systems.	L4
5	**Develop** a simple AI-assisted renewable energy monitoring or energy storage model using simulated or real-time IoT data.	L4

Table: Mapping Levels of COs to POs / PSOs		
COs	**Program Outcomes (POs)**	**PSOs**

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

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS (TB):

3. Soteris Kalogirou – Artificial Intelligence in Energy and Renewable Energy Systems.
4. Alireza Khaligh – Energy Harvesting and Renewable Energies for Smart Devices.
5. Godfrey Boyle - Renewable Energy: Power for a Sustainable Future.

REFERENCE BOOKS:

5. Time Series Prediction: Forecasting the Future and Understanding the Past by Andreas S. Weigend, Addison-Wesley, 1994.
6. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press, 2016.
7. Artificial Intelligence Techniques in Power Systems by Ashish Rajurkar, Ashok Agarwal, IEEE Press / CRC Press, 2017.

E-RESOURCES:

7. NPTEL:
   a. Renewable Energy Systems (IIT Madras / IIT Kharagpur)
   b. Machine Learning for Engineering Applications

ACTIVITY BASED LEARNING:

1. Identification of problem statement by forming a team.
2. Real world problem solving using group discussion.

IV SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	BSC	24AM2401	Transform and Numerical Techniques	3	0	0	0	60	40	100	03
2	IPCC	24AM2402	Design and Analysis of Algorithms	3	0	2	0	60	40	100	04
3	IPCC	24AM2403	Database Management System	3	0	2	0	60	40	100	04
4	PCC	24AM2404	Theory of Computation and System Software	3	1	0	0	60	40	100	04
5	PCC	24AM2405	Artificial Intelligence	3	0	0	0	60	40	100	03
6	PCC	24AM2406	Full Stack Development	3	0	0	0	60	40	100	03
7	SEC	24AM2407	Skill Enhancement Course -II (Unix and Shell Programming)	1	0	2	0	100	--	100	02
8	AEC	24AM2408	Cognitive and Technical Skills-IV	--	--	--	--	--	--	--	Pass /Fail
			Total	19	1	06	0				**23**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, BSC-Basic Science, AEC-Ability Enhancement Course, L – Lecture, T – Tutorial, P – Practical, J– Project, C – No. Of Credits

TRANSFORMS AND NUMERIAL TECHNIQUES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 24AM2401	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–J	: 3–0–0–0		

Course Learning Objectives:

This Course will enable students to:

1. **Apply** their knowledge of Laplace transforms and inverse Laplace transforms to proficiently solve linear ordinary differential equations with constant coefficients, facilitating the analysis and modelling of complex systems.
2. **Analyze** periodic functions using Fourier series, assessing the convergence properties and precision of the series expansion, thereby enhancing their ability to understand and manipulate periodic phenomena.
3. **Utilize** complex exponential form, Fourier transforms of basic functions, and Fourier sine and cosine transforms to solve problems involving Fourier integrals, developing proficiency in applying these techniques to various mathematical scenarios.
4. **Employ** numerical methods, including Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods, to solve differential equations and effectively analyze dynamic systems, enabling them to model real-world phenomena and make accurate predictions**.**
5. **Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to effectively solve different types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations, enhancing their problem-solving skills in the context of differential equations and their applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.

5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding

UNIT – I: Laplace Transform and Inverse Laplace Transform	**09 Hours**

Laplace Transforms of Elementary functions (without proof), ***(Text Book-1: Chapter 6: 203 to 207).***

Laplace Transforms $e^{at}f(t), t^{n}f(t)$ and $\frac{f(t)}{t}$ of Periodic functions, Unit step function and impulse functions

(Text Book-1: Chapter 6:208-230).

Inverse Laplace Transforms- By the method of Partial Fractions, Logarithmic and Trigonometric functions, Convolution Theorem, Inverse Laplace transform using Convolution Theorem ***(Text Book-1: Chapter 6: 238)***. Solution to Differential Equations by Laplace Transform.

(Text Book-1: Chapter 238-242).

UNIT – II: Fourier Series	**09 Hours**

Periodic Functions, Trigonometric Series

(Text Book-1: Chapter 11: 495).

Fourier series Standard function, Functions of any Period 2L, Even and Odd functions, Half-range Expansions. ***(Text Book-1: Chapter 11: 483-492)***

Practical Harmonic analysis (calculate average power and RMS values of periodic waveforms)

UNIT – III: Fourier Transform	**06 Hours**

Calculation of Fourier integrals using complex exponential form ***(Text Book-1: Chapter 11: 510).***

Fourier transform of basic functions ***(Text Book-1: Chapter 11: 510-516).*** Fourier sine and cosine transforms. ***(Text Book-1: Chapter 11: 518-522).***

UNIT – IV: Numerical Methods for Solving Ordinary Differential Equations	**07 Hours**

Euler's Method-Basic principles of Euler's method for solving first-order ODEs ***(Text Book-1: Chapter 1:10-12).***

Runge-Kutta 4th order ***(Text Book-1: Chapter 21:904).***

Multistep Methods-Explanation of multistep methods (Adams-Bashforth, Adams–Moulton Methods) ***(Text Book-1: Chapter 21:911- 913).***

Second-Order ODE. Mass–Spring System (Euler Method, Runge–Kutta Methods) ***(Text Book-1: Chapter 21:916-918).***

UNIT – V: Numerical Methods for Partial Differential Equations	**08 Hours**

Classification of PDEs (elliptic, parabolic, hyperbolic), ***(Text Book-1: Chapter 21:922-923).*** Finite Difference Methods (Laplace and Poisson Equations), Derivation of finite difference approximations ***(Text Book-1: Chapter 21:923-927).*** Crank–Nicolson Method ***(Text Book-1: Chapter 21:938-941).*** Method for Hyperbolic PDEs ***(Text Book-1: Chapter 21:943-945).***

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Apply** Laplace transforms and inverse Laplace transforms to solve linear ordinary differential equations with constant coefficients, demonstrating proficiency in system analysis and modelling.	L3
2	**Analyze** periodic functions using Fourier series and evaluate the convergence properties and precision of the series expansion.	L2 & L3
3	**Solve** problems involving Fourier integrals by applying complex exponential form, Fourier transforms of basic functions, and Fourier sine and cosine transforms.	L3
4	**Utilize** numerical methods such as Euler's Method, Runge-Kutta 4th order, Adams-Bashforth, and Adams-Moulton Methods to solve differential equations and analyze dynamic systems	L2 & L3
5	**Apply** finite difference methods, including the Crank-Nicolson method and appropriate techniques for hyperbolic PDEs, to solve various types of partial differential equations (PDEs) such as elliptic, parabolic, and hyperbolic equations.	L3

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1					1					
CO2	3	2	2						1					

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

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 2015, 10th Edition, Wiley India.

REFERENCE BOOKS:

1. Higher Engineering Mathematics, B.S. Grewal, 2015, 43rd Edition, Khanna Publishers.
2. Higher Engineering Mathematics, John Bird, 2017, 6 th Edition, Elsevier Limited.

E-Resources:

1. https://nptel.ac.in/courses/111106139
2. https://nptel.ac.in/courses/111101164
3. https://nptel.ac.in/courses/111105038

****

DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	:	24AM2402	**Credits**	:	04
Hours / Week	:	05 Hours	**Total Hours**	:	39(Th)+26(P) Hours
L–T–P–J	:	3–0–2–0			

Course Learning Objectives:

This Course will enable students to:

1. **Analyze** the non-recursive and recursive algorithms and to represent efficiency of these algorithms in terms of the standard Asymptotic notations.
2. **Acquire** the knowledge of Brute Force and Divide and Conquer techniques to design the algorithms and apply these methods in designing algorithms to solve a given problem.
3. **Master** the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs.
4. **Learn** Greedy method and dynamic programming methods and apply these methods in designing algorithms to solve a given problem.
5. **Understand** the importance of Backtracking and Branch and Bound algorithm design techniques to solve a given problem.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.
6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT – I	**08 Hours**

INTRODUCTION:
What is an Algorithm? Fundamentals of Algorithmic Problem Solving.
(Text Book-1: Chapter 1: 1.1 to 1.2)
FUNDAMENTALS OF THE ALGORITHMS EFFICIENCY:
Analysis Framework, Asymptotic Notations and Standard notations and common functions ***(Text Book-2: Chapter 3: 3.1, 3.2)***,
Mathematical Analysis of Non-recursive and Recursive Algorithms,
(Text Book-1: Chapter 2: 2.1, 2.3, 2.4,)

UNIT – II	**08 Hours**

BRUTE FORCE:
Background, Selection Sort, Brute-Force String Matching. TSP
(Text Book-1: Chapter 3: 3.1, 3.2)
DIVIDE AND CONQUER:
General method, Recurrences: The substitution method, The recursion-tree method, The master method.
(Text Book-2: Chapter 4: 4.4, 4.5),
Merge sort, Quick sort, Binary Search, Multiplication of large integers,
Case study: Strassen's Matrix Multiplication.
(Text Book-1: Chapter 4: 4.1 to 4.3, 4.5)

UNIT – III	**06 Hours**

DECREASE & CONQUER:
General method, Insertion Sort, Graph algorithms: Depth First Search, Breadth First Search, Topological Sorting
TRANSFORM AND CONQUER:
Case study: Heaps and Heap sort.

TIME AND SPACE TRADEOFFS:
Input Enhancement in String Matching: Horspool's algorithm, Hashing: Open and Closed hashing.
(Text Book-1: Chapter 5: 5.1 to 5.3, Chapter 6: 6.3 to 6.4, Chapter 7: 7.2 to 7.3)

UNIT – IV	**9 Hours**

GREEDY TECHNIQUE:
General method of Greedy technique, Single-Source Shortest Paths: General method, The Bellman-Ford algorithm, Single-Source Shortest Paths in DAGs, Dijkstra's Algorithm
(Text Book-2: Chapter 24: 24.1 to 24.3).
Minimum Spanning Trees: Prim's Algorithm, Optimal Tree problem: Huffman Trees;
Case study: Kruskal's Algorithm. Fractional Problem
(Text Book-1: Chapter 9: 9.1, 9.2, 9.4).
DYNAMIC PROGRAMMING:
General method, The Floyd-Warshall Algorithm, Johnson's algorithm for sparse graphs ***(Text Book-2: Chapter 25: 25.1 to 25.3)***,
The Knapsack problem ***(Text Book-1: Chapter 8: 8.4).***

UNIT – V	**08 Hours**

LIMITATIONS OF ALGORITHMIC POWER
P, NP and NP-complete problems ***(Text Book-1: Chapter 11: 11.3***)

BACKTRACKING:
General method, N-Queens problem, Subset-sum problem.
(Text Book-1: Chapter 12: 12.1)
BRANCH AND BOUND:
General method, Travelling Salesman problem, Approximation algorithms for TSP.
Case study: Knapsack Problem.
(Text Book-1: Chapter 12: 12.2, 12.3)

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Exemplify** the algorithm design techniques and standard Asymptotic notations. Analyze non-recursive and recursive algorithms to obtain worst- case running times of algorithms using asymptotic analysis	L3
2	**Interpret** the brute-force, divide-and-conquer paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.	L3
3	**Demonstrate** the Decrease and Conquer, Transform and Conquer algorithm design techniques and analyze the performance of these algorithms.	L3
4	**Identify and interpret** the greedy technique, dynamic-programming paradigm as to when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms and analyze them	L3
5	**Illustrate** the Backtracking, Branch and Bound algorithm design paradigms and explain when an algorithmic design situation calls for it. Recite algorithms that employ these paradigms. **Summarize** the limitations of algorithmic power.	L3

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**

	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3											2	2
CO2	3	3	2										2	2
CO3	3	3											1	2
CO4	3	3	2										2	2
CO5	3	3											2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Anany Levitin, "Introduction to the Design & Analysis of Algorithms", 2nd Edition, Pearson Education, 2011.
2. Thomas H. Cormen, Charles E.Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rd Edition, PHI, 2014.

REFERENCE BOOKS:

1. Horowitz E., Sahni S., Rajasekaran S, "Computer Algorithms", Galgotia Publications, 2001.
2. R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, "Introduction to the Design and Analysis of Algorithms A Strategic Approach", Tata McGraw Hill, 2005.

E-Resources:

1. https://nptel.ac.in/courses/106/101/106101060/
2. http://cse01-iiith.vlabs.ac.in/
3. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
4. https://www.coursera.org/specializations/algorithms

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

LABORATORY EXPERIMENTS

Total Contact Hours: 26

Following are experiments to be carried out using either C programming language or Object-oriented programming language:

1. Apply divide and conquer method and Design a C program to implementation of Binary Search algorithm.
2. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Demonstrate this algorithm using Divide-and-Conquer method.
3. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Demonstrate this algorithm using Divide-and-Conquer method.
4. Incorporate the array data structure and demonstrate whether a given unweighted graph is connected or not using DFS method.
5. Implement the graph traversal technique using BFS method to print all the nodes reachable from a given starting node in an unweighted graph.
6. Compute the Transitive Closure for a given directed graph using Warshall's algorithm.
7. For a given weighted graph, construct an All-Pairs Shortest Paths problem using Floyd's algorithm and implement this algorithm to find the shortest distance and their shortest paths for every pair of vertices.
8. Implement 0/1 Knapsack problem using Dynamic Programming Memory Functions technique
9. Find Minimum Cost Spanning Tree for a given weighted graph using Prim's and Kruskal's algorithm.
10. From a given vertex in a weighted connected graph, determine the Single Source Shortest Paths using Dijkstra's algorithm.
11. Mini project proposal should be submitted and Implementation should be done based on the problem stated in the proposal

Open ended experiments

1. Implement Fractional Knapsack problem using Greedy Method.
2. Implement N-Queens problem using Backtracking technique.
3. implementation of Travelling Sales man problem using Dynamic programming

****

DATABASE MANAGEMENT SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 24AM2403	**Credits**	: 04
Hours / Week	: 05 Hours	**Total Hours**	: 39(Th)+26(P) Hours
L–T–P–J	: 3–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. **Acquire** the concept of databases, Entity-Relationship Model and relational model for creating and designing databases for the real-world scenario.
2. **Develop** queries to extract data from the databases using a structured query language.
3. **Differentiate** SQL and NoSQL.
4. **Demonstrate** the operations on MongoDB, Database connectivity with front end and **Optimize** the Database design using Normalization Concepts.
5. **Understand** the importance of Transaction Management, Concurrency control mechanism and recovery techniques.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**10 Hours**

INTRODUCTION TO DATABASE SYSTEMS :

Introduction, Characteristics of the Database Approach, Advantages of using DBMS Approach, Data Models, Schemas, Instances and Data Independence, Three Schema Architecture, various components of a DBMS.

(Text Book-1: Chapter 1: 1.1 to 1.4, 1.6, Chapter 2: 2.1,2.2, 2.4)

ENTITY-RELATIONSHIP MODEL:

Entity Types , Entity Sets , Attributes and Keys, Relationship types, Relationship Sets , Roles and Structural Constraints; Weak Entity Types; ER Diagrams

(Text Book-1: Chapter 7: 7.3, 7.4, 7.5, 7.7).

UNIT – II	**07 Hours**
RELATIONAL MODEL : Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update operations and Dealing with Constraint Violations. ***(Text Book-1: Chapter 3: 3.1 to 3.3).*** **SQL –THE RELATIONAL DATABASE STANDARD:** SQL Data Definition and Data types, Specifying constraints in SQL, Basic Queries in SQL-Data Definition Language in SQL, Data Manipulation Language in SQL; ***(Text Book-1: Chapter 4: 4.1 to 4.4).***	
UNIT – III	**08 Hours**
SQL –THE RELATIONAL DATABASE STANDARD: Additional Features of SQL; Views (Virtual Tables) in SQL; Database Programming Issues and Techniques ; ***(Text Book-1: Chapter 4: 4.5; Chapter 5: 5.1 to 5.4).*** **SQL AND NOSQL DATA MANAGEMENT:** Triggers, Database connectivity using Python, SQL vs NoSQL, Introduction to MongoDB, ***(Text Book-1: Chapter 5: 5.2,5.3***)***(Text Book-2 Chapter 1: 1.1 to 1.5***)	
UNIT – IV	**07 Hours**
NOSQL DATA MANAGEMENT: Data Types, Data Modelling, CRUD Operations. ***(Text Book-2 Chapter 1: 1.1 to 1.5***) **DATABASE DESIGN:** Design Guidelines, Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form; ***(Text Book-1: Chapter 14: 14.1 to 14.5)***	
UNIT – V	**07 Hours**
TRANSACTION MANAGEMENT The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Concurrency Control Mechanisms; Error recovery methods. ***(Text Book-1: Chapter 20: 20.1 to 20.5, Chapter 21: 21.1 to 21.3, Chapter 22: 22.1 to 22.4***)	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	**Use** the basic concepts of database management system in the design and creating database blueprint using E-R model and relational model.	L3
2	**Formulate** SQL and NoSQL queries for building structure and unstructured databases	L3
3	**Demonstrate** database connectivity using vendor specific drivers	L3

4	**Apply** normalization techniques to design relational database management system	L3
5	**Adapt** Transaction Management, concurrency control and recovery management techniques in database management system.	L3

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	2	-	-	-	-	-	2	2	-	-	2	3
CO2	3	2	1	-	3	-	-	-	2	2	-	-	2	3
CO3	2	2	2	-	3	-	-	-	2	2	-	-	2	3
CO4	3	1	2	-	1	-	-	-	2	2	-	-	2	3
CO5	2	1	-	-	-	-	-	-	2	2	-	-	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Elmasri and Navathe, ”Fundamentals of Database Systems“, Seventh Edition, Pearson Education, 2021.
2. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", First Edition, Pearson Education, Inc. 2012.

REFERENCE BOOKS:

1. Raghu Ramakrishnan and Johannes Gehrke, “Database Management Systems”, Third Edition, McGraw-Hill, 2003.
2. Silberschatz, Korth and Sudharshan: “Database System Concepts”, Seventh Edition, Mc-GrawHill, 2019.

3. C.J. Date, A. Kannan, S. Swamynatham: "An Introduction to Database Systems", Eight Edition, Pearson Education, 2012.

E-Resources:

1. http://nptel.ac.in/courses/106106093/
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-830-database-systems-fall-2010/lecture-notes/
3. http://agce.sets.edu.in/cse/ebook/DBMS%20BY%20RAGHU%20RAMAKRISHNAN.pdf
4. http://iips.icci.edu.iq/images/exam/databases-ramaz.pdf
5. https://db-class.org/
6. https://www.w3schools.com/mongodb/

Activity Based Learning (Suggested Activities in Class)

1. Database designing and data extraction using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

LABORATORY EXPERIMENTS

Total Contact Hours: 26 Following are experiments to be carried out using either oracle or mysql, Mongo Db .

1. Design any database with at least 3 entities and establish proper relationships between them. Draw suitable ER/EER diagrams for the system. Apply DCL and DDL commands.
2. Design and implement a database and apply at least 10 Different DML Queries for the following task.
    a. For a given input string display only those records which match the given pattern or a phrase in the search string. Make use of wild characters and like operators for the same. Make use of Boolean and arithmetic operators wherever necessary
3. Write SQL statements to join table and retrieve the combined information from tables.
4. Execute the Aggregate functions count, sum, avg, min, max on a suitable database. Make use of built in functions according to the need of the database chosen .
5. Retrieve the data from the database based on time and date functions like now(), date(), day(), time() etc., Use of group by and having clauses.
6. Write and execute database trigger. Consider row level and statement level triggers.
7. Write and execute program to perform operations on MongoDb Database.
8. Write and execute program to perform CRUD operations.

Open Ended Experiments

1. Consider the Table "employees", write a SQL query to remove all the duplicate emails of employees keeping the unique email with the lowest employee id,return employee id and unique emails .

table: employees

Employee id	Employee_name	Email id
101	Liam Alton	li.al@abc.com
102	Josh Day	jo.da@abc.com
103	Sean Mann	se.ma@abc.com
104	Evan Blake	ev.bl@abc.com
105	Toby Scott	jo.da@abc.com

2. A salesperson is a person whose job is to sell products or services. Consider the table "Sales" [given below]. Write a SQL query to find the top 10 salesperson that have made highest sale. Return their names and total sale amount.

Table: sales

TRANSACTION_ID|SALESMAN_ID|SALE_AMOUNT|

501	18	5200.00
502	50	5566.00
503	38	8400.00
599	24	16745.00
600	12	14900.00

Table: salesman

SALESMAN_ID	SALESMAN_NAME
11	Jonathan Goodwin
12	Adam Hughes
13	Mark Davenport
59	Cleveland Hart
60	Marion Gregory

THEORY OF COMPUTATION and SYSTEM SOFTWARE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Subject Code	: 24AM2404	**Credits**:	04
Hours / Week	: 04 Hours	**Total Hours**:	52 Hours
L–T–P–S	: 3–1–0–0		

Course Learning Objectives:

This course will enable students to:

1. Understand the concept of finite automata.
2. Explore Regular Grammar, Context-Free Language & Grammars and Pushdown Automata
3. Explain the working principles of Turing Machines as computational models and undecidability.
4. Explain the functions of assemblers, loaders, and the compilation process including compiler structure and phases.
5. Design and construct parsers using LL(1) and LR parsing techniques, perform grammar transformations, and handle syntax errors effectively.
6. Generate intermediate code using syntax-directed translation, create three-address code representations, and apply fundamental code optimization techniques. RetryPR

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Incorporate real-world examples and case studies that demonstrate the relevance and application of automata theory in areas such as natural language processing, compilers, and pattern recognition*.
2. ***Problem-Solving Sessions:*** *Organize problem-solving sessions where students can work through challenging problems together*.
3. ***Reflective Learning****: Encourage students to reflect on their problem-solving approaches and discuss the reasoning behind their solutions.*

UNIT – I	**11 Hours**

Part 1: Introduction to Finite Automata. The Central Concepts of Automata Theory, Deterministic Finite Automata, Non deterministic Finite Automata, An application, Finite Automata with Epsilon Transitions.

Part II: Regular Expressions Concept, Relation with Finite Automata, Applications of Regular Expressions, Properties of Regular Languages.

UNIT – II	**10 Hours**

Regular Language Context Free Grammars and Languages: Context Free Grammars, Parse Tree, Applications of Context Free Grammar, Ambiguity in Grammars and Languages, Properties of Context Free Languages, The Pumping Lemma for Context Free Languages, Closure Properties of Context Free Languages.	
UNIT – III	**10 Hours**
Definition of Pushdown Automata, Deterministic Pushdown Automata, The Languages of a PDA, , Introduction to Turing Machines: The Turing Machines, Undecidability A language that is not Recursively Enumerable, An Undecidable Problem That is RE, Post Correspondence Problem	
UNIT – IV	**8 Hours**
Introduction to System Software, ASSEMBLERS Introduction to System Software, ASSEMBLERS: Basic assembler functions: A simple assembler, Machine dependent assembler features, Machine independent assembler features: Literals, Symbol-defining statements, Expressions, Program blocks, Basic loader functions: Design of an Absolute Loader, A Simple Bootstrap Loader, Introduction: Language Processors, Structure of compiler, Phases of Compiler.	
UNIT – V	**13 Hours**
COMPILERS Part I: LEXICAL AND SYNTAX ANALYSIS: Role of lexical Analyzer, Specification of Tokens, Lexical Analyzer generator Lex. SYNTAX ANALYSIS I: Role of Parser, Syntax error handling, Error recovery strategies, Writing a grammar: Eliminating ambiguity, Left recursion, Left factoring. Part II: Top down parsing: Recursive Descent Parsing, First and follow, LL (1), –Bottom up parsing: Shift Reduce Parsing, Introduction to LR parsing Simple LR: Why LR Parsers, Items and LR0 Automaton, The LR Parsing Algorithm. SYNTAX-DIRECTED TRANSLATION: Syntax-Directed Definitions: Inherited and Synthesized Attributes, Evaluation orders for SDDs: Dependency graphs, Ordering the evaluation of Attributes,Three address code: Quadruples, Triples, indirect triples, Basic Blocks, Optimization of Basic Blocks, The Code Generation Algorithm, Peephole optimization	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Analyze and formulate finite Automata, Regular Expressions for Text Processing	L3
2	Demonstrate Understanding of Context-Free Grammars and Pushdown Automata	L3
3	Apply Turing Machines to explore Undecidability and Computational Limits	L4
4	Construct lexical analyzers and parsers (LL(1), LR) with grammar transformation capabilities.	L4
5	Implement syntax-directed translation, generate intermediate code, and perform basic code optimization.	L3

Mapping Levels of COs to POs / PSOs														
COs				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Developmen	Conduct Investigatio	Modern Tool Usage	The Engineer	Environmen t and	Ethics	Individual & Team Work	Communica tion	Project managemen	Life-long Learning	Cognitive Outcome	Skill & Design
CO1	3	3	1		2	2						1	2	1
CO2	3	3	2		2						1	1	3	2
CO3	3	3	2	2	3					2	1	3	3	3
CO4	3	2	1		3						1	2	2	2
CO5	3	2	3	3	3						2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Esparza, Javier, and Michael Blondin. *Automata theory: An algorithmic approach*. MIT Press, 2023.
2. Pettorossi, Alberto. *Automata Theory and Formal Languages: Fundamental Notions, Theorems, and Techniques*. Springer Nature, 2022.
3. "System Software: An Introduction to Systems Programming" by Leland L. Beck and D.Manjula was the 3rd edition, published in 2020.
4. "Compilers: Principles, Techniques, and Tools" by Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman (commonly known as the "Dragon Book") 2nd Edition 2020.

REFERENCE BOOKS:

1. K.L.P. Misra and N. Chandrashekaran. Theory of Computer Science- Automata, Languages and Computation, 3rd Edn. PHI, New Delhi, 2007
2. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2013
3. J.E. Hopcroft , R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edn. Pearson Education ,2013
4. Computation, 3rd Edn. PHI, New Delhi, 2007
5. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2013
6. V. Raghavan, Principles of Compiler Design||, Tata McGraw Hill Education Publishers, 2010.

7. Keith D Cooper and Linda Torczon, Engineering a Compiler, Morgan Kaufmann Publishers Elsevier Science, 2004. 3. D.M.Dhamdhere, Systems Programming and operating systems, Second Revised edition, Tata McGraw Hill.

TEXTBOOKS:

E-Resources:

- https://archive.nptel.ac.in/courses/111/103/111103016/
- https://www.youtube.com/watch?v=58N2N7zJGrQ&list=PLBlnK6fEyqRgp46KUv4ZY69yXmpwKOIev
- https://www.geeksforgeeks.org/theory-of-computation-automata-tutorials/
- Compiler Design - Course (nptel.ac.in)

Activity Based Learning (Suggested Activities in Class)

Hands on activity, Quiz, Seminar

ARTIFICIAL INTELLIGENCE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 24AM2405	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students to:

1. To provide a strong foundation of fundamental concepts in artificial intelligence.
2. To provide a basic exposition to different types of searching in Artificial intelligence.
3. To provide different knowledge representation, reasoning, and learning techniques.
4. To distinguish the different types of Experts Systems.
5. To design the Expert System based on the concepts of knowledge representation, searching and Reasoning on various applications using modernised AI tools.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**

Introduction: Artificial Intelligence, AI Problems, AI Techniques: supervised learning, unsupervised learning, and reinforcement learning, Different Types of Agents: Simple reflex agents, Model-based reflex agents, Goal-based agents and Utility-based agents; Environment, Problem Space and Search, Defining the Problem as a State Space Search, Problem Characteristics.

Textbook 2: Chapter 1,2

UNIT – II	**08 Hours**

Basic Search Techniques: Solving problems by searching; issues in the design of search programs; uniform search strategies: Breadth first search, Depth first search, Depth limited search, Bidirectional search, Best First search. Textbook 3: Chapter 3, 4; Textbook 2: Chapter 2, 3
UNIT – III **08 Hours**
Special Search Techniques: Heuristic Search, greedy best first search, A* search, AO*Algorithm, Hill climbing search, Simulated Annealing search, Adversarial search, Minimax search, Alpha, beta pruning, Genetic Algorithm. Knowledge Representation: Procedural Vs Declarative Knowledge, Approaches to Knowledge Representation, Forward Vs Backward Reasoning. Textbook 1: Chapter 5; Textbook 2: Chapter 3, 6
UNIT – IV **07 Hours**
Symbolic Logic: Propositional Logic, First Order Predicate Logic: Representing Instance and isa Relationships. Reasoning: Introduction to Monotonic Reasoning and Non-Monotonic Reasoning. Statistical Reasoning: Bayes Theorem, Certainty Factors, Bayesian Networks, Dempster-Shafer Theory. Textbook 2: Chapter 7,8
UNIT – V **08 Hours**
Experts Systems: Overview of an Expert System, Structure of an Expert Systems, Different Types of Expert Systems: Rule Based, Model Based, Case Based and Hybrid Expert Systems, Knowledge Building System Tools, Expert System Shells and Fuzzy Expert systems. Textbook 1: Chapter 8,9; Textbook 2: Chapter 20,22

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Summarize** the basic concepts of Artificial Intelligence, AI principles, AI Task domains and applications.	L2
2	**Apply** the different types of searching and knowledge representations techniques to solve the AI problems.	L3
3	**Make use of** the different types of reasoning such as Monotonic Reasoning, Non-Monotonic Reasoning and Statistical Reasoning techniques for making analysis and predictions.	L4

4	**Differentiate** the various types of Experts Systems such as Rule Based, Model Based, Case Based, Hybrid and Fuzzy Expert Systems.	L4
5	**Design** the Expert System based on the concepts of knowledge representation, searching and Reasoning on various applications using modernised AI tools.	L5

Mapping Levels of COs to POs / PSOs														
				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
COs	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	2				2	1			1	1			2	2
CO2	3	2			2	1			1	1			2	2
CO3	3	2			2	1			1	1			2	2
CO4	3	2			2	1			1	1			2	2
CO5	3	3	2		2	1			1	1			2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Artificial Intelligence, George F Luger, Sixth Edition, Pearson Education Publications, 2014.
2. Artificial Intelligence, Elaine Rich and Knight, McGraw-Hill Publications, 2010.

REFERENCE BOOKS:

1. Artificial Intelligence: A modern Approach, Russell and Norvig, Printice Hall, 2010.

2. Introduction To Artificial Intelligence & Expert Systems, Patterson, PHI.
3. Multi Agent systems- a modern approach to Distributed Artificial intelligence, Weiss. G, MIT Press.
4. Artificial Intelligence: A modern Approach, Russell and Norvig, Printice Hall.

E-Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs56/preview
2. https://onlinecourses.nptel.ac.in/noc23_ge40/preview

Activity Based Learning (Suggested Activities in Class)

1. Flipped Class Activity on Searching techniques.
2. Problem Solving and Discussion.
3. Role Play
4. Mini Project

****

FULL STACK DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 24AM2406	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students to:

1. To demonstrate the ability to design basic web pages using HTML and CSS
2. To utilize advanced CSS methods to design the web page, background and components.
3. To experiment with Bootstrap, Javascript and DOM for to create dynamic webpages.
4. To build web pages with the ReactJS and NodeJS as front end and back end.
5. To Model a complete web-page with mangoDB CRUD operations.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**

Introduction to HTML: HTML syntax, HTML elements, HTML document structure, CSS selectors, properties, and values, Build your first web page, Semantics of HTML, Block and inline elements, Hyperlinks, Lists, Tables, Forms, Create a basic multi page website, Cascading effect, Advanced selectors, CSS resets, Positioning

UNIT – II	**08 Hours**

Introduction to CSS: CSS syntax ,Cascading effect, Specificity, Combining and layering selectors, Backgrounds and gradients, CSS resets, The box model, Positioning with floats, Creating a grid structure, Precise Positioning, Basic website clone.

UNIT – III	08 Hours
Bootstrap and JavaScript Basics: Introduction to Bootstrap, Bootstrap Basics ,Bootstrap Grids, Bootstrap Themes, Bootstrap CSS, Bootstrap JS, Fundamentals Of JavaScript, Fundamentals of jQuery, Fundamentals of Ajax Development, Document Object Model, DOM Manipulation, DOM Events, JavaScript Libraries (jQuery and Underscore), Simple HTML CSS JavaScript Project with AJAX.	
UNIT – IV	**08 Hours**
ReactJS and NodeJS Development: Introduction and Foundation, Node Projects, Working with shrink-wrap to lock the node modules versions, Working with asynchronous programming, Building a HTTP Server with Node.JS using HTTP APIs, React Components, React State and Props, React Event Handling, Routing in React React flux, Styling React.	
UNIT – V	**08 Hours**
MongoDB Development: Introduction to NoSQL databases, MongoDB A Database the Modern Web, CRUD Operations in MongoDB, Indexing and Aggregation, Replication and Sharding, Developing Java and Node JS Application with MongoDB. LAMP/WAMP/XAMP	

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO1	3	3	3		3				-						
CO2	3	3	3		3				-						
CO3	2	2	2		3	2			-		3	2	2	2	2
CO4	3	3	3		3	2			-		3	2	2	2	2
CO5	3	1	3		3	2			-		3	2	2	2	2

Text Books:

1. Riaz Ahmed-Full Stack Web Development For Beginners,2021
2. Chris Northwood - The Full Stack Developer: Your Essential Guide to the Everyday Skills
3. Edwin Ross Torres -Full Stack Web Development,2020

Reference Books:

1. Ahmed Bouchefra,- Full Stack Development with Angular and GraphQL, 2022
2. David Choi -Full-Stack React, TypeScript, and Node,2020.
3. Valerio De Sanctis-ASP.NET Core 5 and Angular,2021

E-Resources:

1.Manu Sharma, Full Stack Development with MongoDB, bpb publishers
2. Apress Modern Full-Stack Development by Zammetti

SKILL ENHANCEMENT COURSE-II
UNIX AND SHELL PROGRAMMING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – IV

Course Code	: 24AM2407	**Credits**	: 02
Hours / Week	: 03	**Total Hours**	: 39 Hours
L–T–P–S	: 1–0–2–0		

Course Learning Objectives:

This Course will enable students to:

1. Understand the basic UNIX process structure and the UNIX file system.
2. Understand the roles of unix developers / systems programmers
3. Good knowledge of simple UNIX filters.
4. Familiar with pipes and redirection, imagine the UNIX environment.
5. Practice various commands related to Signals, filter parameters and options,
6. Differentiate shell scripting and commands practice with various options.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture method, but different *type of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

UNIT I	**4 Hour**

Introduction to operating system and Unix: computer system organization, Computer system architecture, The Structure of the
UNIX Operating System, Linux Kernel, Applications, Understanding Linux. Unix vs. Linux, Downloading and Installing Oracle Virtual Box, Linux Distributions. System login. Directories Found On Linux Systems. Manipulating Files And Directories**:** cp – Copy files and directories, mv – Move/rename files and directories, mkdir – Create directories, rm – Remove files and directories, ln – Create hard and symbolic links.

(Text Book-3: chapter1; Text Book-1: Chapter 4)

UNIT II	**6 Hour**

Permissions, file attributes, ownership**:** Exploring system with ls command, Owners, Group Members, And Everybody Else; Reading, Writing, And Executing; File attributes and chmod – Change File Mode, Changing Identities, su and sudo commands; chown, passwd. Process and signals. Environment and bashrc. Text editors: vi and nano ***(Text Book-1: Chapter 3,7,11)***

UNIT III	5 Hour
grep, awk and ipcs: awk, awk and regular expression, pipe line and specifying patterns, actions, grep command and its options for string handling. Writing your first script: Script File Format, Executable Permissions, Script File Location ***(Text Book-2: Chapter 21; Text Book-1: Chapter 19)***	
UNIT - IV	**6 Hour**
Starting with Linux Shells: Using Shell Variables , Special Variables , Using Shell Arrays, — Shell Basic Operators ,Shell Decision Making, Unix — Shell Loop Types, Shell Input/Output Redirections, — Shell Functions. ***(Text Book-2: Chapter***	
UNIT - V	**5 Hour**
Some advanced commands: File System Check (fsck and xfs_repair), System Backup (dd Command), Network File System (NFS), SSH command and its options, Download files with URLs (wget), curl and ping commands, File transfer commands (ftp, scp etc.).	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Elucidate** basics organization of system and operating system.	L2
2	**Demonstrate** the manipulation of file system, directories and file attributes using commands.	L3
3	**Interpret** the inter process communication.	L4
4	**Implement** shell scripts for running a program or creating a program environment.	L4
5	**Use** advance commands to manage filesystem and access remote connection	L5

Mapping Levels of COs to POs / PSOs														
				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
COs	Engineering knowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer and society	Environment and sustainability	Ethics	team work	Communication	Project management and finance	Life-long learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3							2					

CO2	3	3	2						2	2				
CO3	3	3							2	2				
CO4	3	3	2						2	2				
CO5	3	3							2	2				

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Suggested Experiments :

1. Use the date and who commands in sequence (in one line) such that the output of date will display on the screen and the output of who will be redirected to a file called myfile2.
2. List the top 10 files where the filename begins with a,e,i,o,u and Find and list all the binary files in your environment
3. Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory
4. Write a shell script that computes the gross salary of a employee according to the following rules:
   i) If basic salary is < 1500 then HRA =10% of the basic and DA =90% of the basic.
   ii) If basic salary is >=1500 then HRA =Rs500 and DA=98% of the basic The basic salary is entered interactively through the key board.

TEXT BOOKS:

1. Sarwar, Syed Mansoor, and Robert M. Koretsky. UNIX: the textbook. Chapman and Hall/CRC, 2016.
2. Rosen, Kenneth H., Douglas A. Host, Rachel Klee, and Richard R. Rosinski. *UNIX: the complete reference*. McGraw-Hill, Inc., 2006.
3. Operating System Concepts Essentials, Binder Ready Version - By Abraham Silberschatz, Peter B. Galvin, Greg Gagne - 2014

REFERENCE BOOKS:

1. Ebrahim, Mokhtar, and Andrew Mallett. Mastering Linux Shell Scripting: A practical guide to Linux command-line, Bash scripting, and Shell programming. Packt Publishing Ltd, 2018.

E-Resources:

1. https://docs.kernel.org/
2. https://linux.die.net/
3. https://bjpcjp.github.io/pdfs/devops/linux-commands-handbook.pdf
4. https://www.unixtutorial.org/basic-unix-commands

Activity Based Learning (Suggested Activities in Class)

1. Mini projects using Linux.
2. Exploration of Linux distros

****

V SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	IPCC	24AM3501	Machine Learning	3	0	2	0	60	40	100	04
2	IPCC	24AM3502	Operating Systems	3	0	2	0	60	40	100	04
3	PCC	24AM3503	Natural Language Models	3	1	0	0	60	40	100	04
4	PCC	24AM3504	Image Processing and Computer Vision	3	0	0	0	60	40	100	03
5	HSMC	24AM3505	Innovation and Entrepreneurship	2	0	0	0	60	40	100	02
6	SEC	24AM3506	Skill Enhancement Course –III (Cloud Computing)	1	0	2	0	100	--	100	02
7	PEC	24AM35X X	Professional Elective Course -I	2	0	0	2	100	--	100	03
8	AEC	24AM3507	Cognitive and Technical Skills-V	-	-	-	-	--	--	--	Pass/Fail
			Total	17	1	6	2				**22**

HSMC-
Humanities and Social Sciences including Management Courses, IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, AEC-Ability Enhancement Course, L – Lecture, T – Tutorial, P – Practical, J-Project, C – No. Of Credits

Professional Elective -1 (PEC-1) Offering										
Sl. No	**Course Type**	**Course Name**	**Teaching Hours / Week**				**Examination**			
			Lecture	**Tutorial**	**Practica**	**Project**	**CIE Marks**	**SEE Marks**	**Total Marks**	**C**
			L	**T**	**P**	**J**				
1	PEC	Ethics, Safety & Governance of Agentic AI	2	0	0	2	100	--	100	**03**
2	PEC	Intelligent Speech Processing Systems	2	0	0	2	100	--	100	**03**
3	PEC	Foundations of Robotics and Intelligent Systems	2	0	0	2	100	--	100	**03**
4	PEC	Explainable AI	2	0	0	2	100	--	100	**03**
5	PEC	Cognitive Linguistics	2	0	0	2	100	--	100	**03**
DLI INTEGRATED										
5	DLI/MOOC	Accelerated Computing								

MACHINE LEARNING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 24AM3501	**Credits**	: 04
Hours / Week	: 05 Hours	**Total Hours**	: 39(T)+26(P) Hours
L–T–P–S	: 3–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. **Summarize** the basic concepts and different types of Machine Learning including Supervised learning, Unsupervised and Reinforcement learning Techniques.
2. **Explore** and analyze the mathematics behind Machine Learning algorithms to gain a solid understanding of probability density functions, basics of sampling theorem and estimation of the maximum likelihood.
3. **Make Use of** the Supervised Machine Learning techniques for solving appropriate real-world applications.
4. **Make Use of** the Unsupervised Machine Learning techniques , Feature Engineering and Dimensionality Reduction techniques for solving appropriate real-world applications.
5. **Evaluate** the performance of Machine Learning algorithms using appropriate metrics such as accuracy, precision, recall, FI Score and Make Use of the different Optimization and Regularization Techniques for improving the model performance.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**6 Hours**

INTRODUCTION TO MACHINE LEARNING

Well-posed learning problems, Designing a Learning system. Introduction to AI, Machine learning, and Deep learning with applications. Types Of Learning: Supervised, Unsupervised, And Reinforcement Learning. Perspective and Issues in Machine Learning.

Classical Paradigm of Solving Learning Problems, The Learning Problems--Classes and Types of Learning, Fundamental of Statistical Learning And Its Framework. Introduction to Feature Representation and Extraction.

Text-2-Chapter 1

UNIT – II	9 Hours
MATHEMATICS FOR MACHINE LEARNING Introduction To Statistics And Probability: Probability concepts - Axioms of probability, Notion of random variables, PMF, PDFs, CDFs. Two Random Variables, Pairs of Discrete Random Variables, Joint Probability, Conditional Probability, Bayes Theorem, Different Distributions, Univariate And Multivariate Gaussian Distribution, PDF, MLE, Motivation, Estimating Hypothesis Accuracy, Basics of the Sampling Theorem, General Approach For Deriving Confidence Intervals, Difference in the Error of Two Hypotheses, Comparing Learning Algorithms. Text-2-Chapter 2	
UNIT – III	**8 Hours**
SUPERVISED LEARNING Introduction to Supervised Learning: Introduction to Classification, Naive Bayes Classification Binary and Multi-Class Classification, Decision Trees and Random Forest, Regression (Methods Of Function Estimation) -- Linear Regression And Nonlinear Regression, Logistic Regression, Introduction to Kernel Based Methods of Machine Learning: K-Nearest Neighborhood, Kernel Functions, SVM, Introduction to Ensemble-Based Learning. Text-2-Chapter 2	
UNIT – IV	**8 Hours**
UNSUPERVISED LEARNING Introduction To Unsupervised Learning, Clustering (Hard and Soft Clustering) Hierarchal Clustering: K-Means, Fuzzy C-Means (FCM) Algorithm, Gaussian Mixture Models (GMM), Expectation Maximization Algorithm, Feature Engineering In Machine Learning, Dimensionality Reduction, Linear Discriminant Analysis And Principal Component Analysis. Text-2-Chapter 3	
UNIT – V	**8 Hours**
MODEL SELECTION Machine Learning Model Validation - Confusion Matrix, Accuracy, Precision, F Score, Cost Function, Machine Learning Optimization Algorithms: Gradient Descent, Stochastic GD. Regularization: Normalization and Standardization, Overfitting, Underfitting, Optimal Fit, Bias, Variance, Cross-Validation. Text-2-Chapter 5	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Describe** the basic concepts and different types of Machine Learning techniques including Supervised learning, Unsupervised learning and Reinforcement learning techniques and design Well Posed Learning system for solving the problem.	L2
2	Explore and **Analyze** the mathematics behind Machine Learning algorithms to gain a solid understanding of probability density functions, the basics of sampling theorem and estimating the	L4

	maximum likelihood.	
3	**Apply** the learned concepts of machine learning to interpret the Supervised learning algorithms including regression and classification problems.	L3
4	**Apply** Unsupervised Machine Learning Algorithms such as Hard and Soft clustering and Feature Engineering and Dimensionality Reduction techniques for solving appropriate real-world applications.	L3
5	**Evaluate** the performance of Machine Learning algorithms using appropriate metrics such as accuracy, precision, recall, FI Score and Make Use of the different Optimization and Regularization Techniques for improving the model performance.	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO 1	2	2	1	-	-	-	-	-	-	-	1	-	2	2
CO 2	2	2	1	-	-	-	-	-	-	-	1	-	2	2
CO 3	3	2	1	1	1	2	1	-	2	1	1	-	2	2
CO 4	3	2	1	1	1	2	1	-	2	1	1	-	2	2
CO 5	3	2	1	1	1	2	1	-	2	1	1	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXTBOOKS:

1. Thomas M. Mitchell, Machine Learning, McGraw- Hill, Inc. New York, ISBN-13: 978-1259096952, 2017.

2. Andreas Muller, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'reilly, 2016.

REFERENCE BOOKS:

1. Ethem Alpaydin, Introduction to Machine Learning (Adaptive Computation and Machine Learning series), The MIT Press; second edition, 2009.
2. U Dinesh Kumar Manaranjan Pradhan, Machine Learning Using Python, Wiley India Pvt. Ltd, 2019.
3. Mark Fenner, Machine Learning with Python for Everyone, Addison-Wesley Professional, 2019.

E-Resources:

1. https://machinelearningmastery.com
2. https://www.knuggets.com/
3. https://www.geeksforgeeks.org/machine-learning-projects/

MOOC's Courses:

1. "Introduction to Machine Learning", NPTEL
2. "Machine Learning for Engineering and Science applications", NPTEL

Activity-Based Learning (Suggested Activities in Class)

1. Workshops/Seminar (ML Projects Based on Python)
2. Quiz

LIST OF LABORATORY/PRACTICAL EXPERIMENTS ACTIVITIES TO BE CONDUCTED
1. Write a Program to Implement the Water-Jug problem using Python. 2. Apply Linear Regression on a given dataset and comment on their efficiency and performance. 3. •Implement linear regression using sklearn library. 4. •Split the obtained dataset into training and testing: 80-20, 70-30 ratio. 5. •Evaluate the model using metrics: Accuracy, Mean Square Error. 6. Apply logistic regression to the given dataset and evaluate the model using a confusion matrix. Download the dataset from the repository and import it as input. Build a Logistic Regression Model and train the dataset. 7. Apply Naïve Bayes Classifier to the given dataset and evaluate the model using a confusion matrix Precision, Recall, F score, and AUC-ROC curve. 8. Use K Nearest Neighbor technique on a given dataset and analyze the performance by changing the value of K. 9. •Implement the KNN algorithm. 10. •Apply KNN model on the dataset and perform testing on unseen dataset. 11. •Change the value of K in KNN and analysis the performance of the model. 12. Build a machine learning model using the Decision Tree algorithm to predict whether a breast tumor is benign or malignant based on various characteristics of the tumor, improving diagnostic accuracy in medical applications. 13. Apply a Random Forest Classifier for classifying breast cancer tumors as benign or malignant based on features like radius and texture. 14. Use Support Vector Machine to perform the classification and regression on given dataset. 15. •Import the dataset and perform the classification using SVM. 16. •Evaluate the model's performance on testing dataset and validation dataset 17. •Use Sklearn's Grid Search CV method for fine tuning of hyper-parameters.

18. Apply K-Means Clustering on the collected dataset.
19. •Implement K-Means clustering using sklearn.
20. •Check for the best k-value.
21. Implement Principal Component Analysis (PCA) for dimensionality reduction and data visualization, demonstrating both from-scratch implementation and sklearn comparison on iris and digits datasets.
22. Implement both simple Perceptron and Multi-Layer Perceptron (MLP) neural networks, demonstrating binary classification with custom implementation and multi-class classification using TensorFlow/Keras.
23. Implement Byte Pair Encoding (BPE) for text processing, a tokenization technique that reduces vocabulary size by iteratively merging the most frequent character pairs in a corpus while retaining important sub word components.

OPERATING SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	24AM3502	**Credits**	:	04
Hours /	:	05 Hours	**Total Hours**	:	39(T)+26(P) Hours

Week

L–T–P–S : 3–0–2–0

Course Learning Objectives:

This course will enable students to:

1. To understand the basic concepts, types, and functions of operating systems.
2. To understand the Processes and Threads.
3. To analyze the process Scheduling algorithms.
4. To understand the concept of Deadlocks and process synchronization.
5. To analyze various Memory management techniques.
6. To understand the various file management approaches.
7. To analyze the disk scheduling algorithms.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I: OS Overview and System Structure	**05 Hours**

Introduction to operating systems and System structures, Computer System organization; Computer System architecture; Operating System structure; Operating System operations;

Different types of operating system: Batch Processing, multi-programmed, time-sharing, real-time, distributed, parallel.

Operating System Services: User and operating System interface; System calls; Types of system calls; operating system structure; Virtual machines.

UNIT – II: Process Management	**09 Hours**

Process Management: Process concept; Process scheduling; Operations on processes.

Threads : Overview; Multithreading models; Threading issues, User and Kernel threads.

Process Scheduling Algorithms: Basic concepts; Scheduling Criteria; Scheduling Algorithms (FCFS, SJF, SRTF, RR, Priority).

UNIT – III: Process Coordination	**09 Hours**

Process Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Deadlocks: Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

UNIT – IV: Memory Management	**08 Hours**
Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. **Virtual Memory Management:** Background; Demand paging; Performance; Page replacement algorithms (FCFS, LRU, Optimal); Allocation of frames; Thrashing.	
UNIT – V File & Disk Management	**08 Hours**
File Systems: File concept, access methods, directory structure, file system structure, allocation methods (contiguous, linked, indexed), free-space management (bit vector, linked list, grouping). Disk Management: Disk structure, Disk scheduling (FCFS, SSTF, SCAN, C-SCAN).	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	**Explain** the basic concept of OS, structures, and different types of OS	L2
2	**Apply** the knowledge of process, threads, and scheduling techniques to manage the system resources.	L3
3	**Make use of** Synchronization and Deadlock Handling techniques to solve the real-world problems.	L3
4	**Analyze** the memory management techniques like paging, segmentation, and page fault handling techniques.	L4
5	**Analyze and distinguish** the file and disk management approaches to manage resources in operating system.	L4

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO 1	3													

CO 2	3	3		2					2	2			2	
CO 3	3	3	3	3					2	2			2	
CO 4	3	3	3	3					2	2			2	3
CO 5	3	3		2									2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Kossiakoff, Alexander, Steven M. Biemer, Samuel J. Seymour, and David A. Flanigan. *Systems engineering principles and practice*. John Wiley & Sons, 2020.
2. Jaeger, Trent. *Operating system security*. Springer Nature, 2022.

REFERENCE BOOKS:

1. Greg Gagne, Abraham Silberschatz, Peter B. Galvin, Operating System Concepts Essentials, 2nd Edition 8th edition, ISBN: 978-1-118-80492-6, Wiley-India, 2013.
2. Operating Systems-Internals and Design Principles, William Stallings, 9th Edition, Pearson Education, ISBN-13: 978013751674, 2021.
3. Andrew S. Tanenbaum, Modern Operating System, 5th Edition. Pearson, 2022.
4. Dhamdhere, D. M. Operating Systems: A Concept Based Approach, 3rd Edition (Indian), ISBN: 9781259005589, McGraw Hill Education, 2017.
5. Fox, Richard. *Linux with operating system concepts*. Chapman and Hall/CRC, 2021.
6. Chakraborty, Pranabananda. *Operating Systems: Evolutionary Concepts and Modern Design Principles*. CRC Press, 2023.

E-Resources:

-NIL-

Activity Based Learning (Suggested Activities in Class)

1. Quiz.
2. Group Discussion.

LABORATORY EXPERIMENTS

Following are experiments to be carried out using either C programming language

1. Write a program for the implementation of various CPU scheduling algorithms (FCFS, SJF, Priority).
2. Write a program for the implementation of various page replacement algorithms (FIFO, Optimal, LRU).
3. Write a program for the implementation of Producer-Consumer problem.
4. Write a program for the implementation of Readers Writers problem.
5. Write a program for the implementation of Banker's algorithm.
6. Write a program to simulate the concept of semaphores.
7. Write a program to simulate the concept of inter process communication.
8. Write a program for the implementation of various memory allocation algorithms (First fit,

Best fit, and Worst fit).

9. Write a program for the implementation of various Disk scheduling algorithms (FCFS, SCAN, SSTF, C-SCAN).

****

NATURAL LANGUAGE MODELS [As per Choice Based Credit System (CBCS) scheme] **SEMESTER – V**			
Subject Code	: 24AM3503	**Credits**:	04
Hours / Week	: 04 Hours	**Total Hours**:	52 Hours
L–T–P–S	: 3–1–0–0		

Course Learning Objectives:

This course will enable students to:

1. To understand the algorithms available for the processing of linguistic information and computational properties of natural languages
2. To conceive basic knowledge on various morphological, syntactic and semantic NLP task
3. To understand machine learning techniques used in NLP,
4. To write programs in Python to carry out natural language processing

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**08 Hours**
Past, present, and future of NLP; Classical problems on text processing; Necessary Math concepts for NLP; Regular expressions in NLP. Parts of Speech and Morphology, Phrase Structure, Semantics and Pragmatics, Corpus-Based Work: Getting Set Up, Looking at Text, Marked-up Data Text processing: lemmatization, stop word, tokenization, stemming, Spelling errors corrections–Minimum edit distance, Bayesian method	
UNIT – II	**08 Hours**
Words & Sentences, N-grams: Simple unsmoothed n-grams; smoothing, backoff, spelling correction using N-grams, Metrics to evaluate N-grams. Parts of Speech tagging: Word classes, POST using Brill's Tagger and HMMs; Information Extraction: Introduction to Named Entity Recognition and Relation Extraction WordNet and WordNet-based similarity measures, Concept Mining using Latent Semantic Analysis	
UNIT – III	**08 Hours**
Sequence to sequence & Language Modelling, Word embedding: skip-gram model, CBOW, GloVe, Language Modelling: Basic ideas, smoothing techniques, Language modeling with RNN and LSTM	
UNIT – IV	**08 Hours**
Case studies on Generative AIs in NLP: History of generative AI, ChatGPT technical overview, Generative pre-trained Transformer – 1, Generative pre-trained Transformer – 2, and Generative pre-trained Transformer – 3.	
UNIT – V	**07 Hours**
Advanced Topics and Hands-on Practices Python libraries supporting NLP; Hands-on Data collection - from social network platforms, pdfs, word files, JSON, HTML Parsing text using regular expression; scraping data from web; Text processing: convert to lowercase, remove punctuation, remove stop words, standardizing text, tokenising, stemming, lemmatising. Applications: Spam detection, consumer complaint classification, Semantic Analyser, Dialogue processing (Chatbots), Text summarization, Text Categorization.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		

1	Demonstrate an understanding of fundamental NLP concepts, including text processing techniques and classical problems in NLP.	L2
2	Analyze and evaluate different NLP methods and algorithms for tasks such as part-of-speech tagging and named entity recognition.	L4
3	Apply NLP techniques to real-world problems, such as spam detection and text summarization, using Python libraries.	L4
4	Compare and contrast advanced NLP models, such as language models using recurrent neural networks (RNNs) and generative pre-trained transformers (GPTs).	L5
5	Synthesize their knowledge of NLP concepts and evaluate the techniques to design and develop their own NLP applications, such as chatbots or text categorization systems.	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO 1	3	3	2										2	1
CO 2	3	3	2						2	2			2	1
CO 3	3	2	1		2				2	2			2	1
CO 4	3	2	2		2				2	2			2	1
CO 5	3	2	2		2				2	2			2	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, "Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World Nlp Systems" - "O'Reilly Media, Inc.", 17 Jun 2020.
2. Transformers for Natural Language Processing: Build innovative deep neural network architectures for NLP with Python, PyTorch, TensorFlow, BERT, RoBERTa, and more - Denis Rothman, Packt Publishing Ltd, 2021

3. Daniel Jurafsky and James H. Martin. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition. Prentice-Hall.
4. Tiwary, U. S., & Siddiqui, T. (2008). Natural language processing and information retrieval. Oxford University Press, Inc.

REFERENCE BOOKS:

1. Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems - Aurélien Géron,Edition 2, O'Reilly Media, 2017.
2. Deep Learning for Natural Language Processing - Palash Goyal, Sumit Pandey, Karan Jain, Apress Berkeley, CA- 2018

E-Resources:

1. https://github.com/topics/nlp-models
2. https://devopedia.org/site-map/browse-articles/natural%20language%20processing
3. https://wisdomml.in/hidden-markov-model-hmm-in-nlp-python/
4. https://spotintelligence.com/2023/06/16/activation-function/
5. https://radimrehurek.com/gensim/models/word2vec.html

Activity Based Learning (Suggested Activities in Class)

1. Better Understanding the concept of Sampling and Semantic Role Labeling Quantization of Speech and using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

IMAGE PROCESSING AND COMPUTER VISION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3504	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students:

1. To understand the algorithms available for the processing of linguistic information and computational properties of natural languages.
2. To conceive basic knowledge on various morphological, syntactic, and semantic NLP tasks.
3. To understand and analyze the fundamental concepts of Computer Vision
4. To understand and analyze the fundamental concepts of Computer Vision.
5. To learn to use deep learning tools and framework for solving real-life problems related to images and signals.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

INTRODUCTION:

Digital Image Fundamentals: Elements of Visual Perception, A Simple Image Model, Sampling and Quantization, Image File Formats, Color Models, In spatial domain: Basic gray level transformations, Histogram processing, using arithmetic/Logic operations, smoothening spatial filters, Sharpening spatial filters. In Frequency domain: Introduction to the Fourier transform and frequency domain concepts, Frequency-domain filters: Low pass filter, High pass filter, Band pass filter, Sharpening frequency domain filters.

(Text1: Chapters 1, 2, 3)

UNIT – II	**08 Hours**

IMAGE SEGMENTATION:

Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region-based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold.

COLOR IMAGE PROCESSING:

Color fundamentals, Color models, Color transformation, Smoothing and Sharpening, Color segmentation.

(Text1: Chapters 6, 10)

UNIT – III	**08 Hours**

MORPHOLOGICAL IMAGE PROCESSING:

Erosion, dilation, opening, closing, Basic Morphological Algorithms: hole filling, connected components, thinning, skeleton.

FEATURE EXTRACTION:

Textural Features, Shape Features, Color Features. Image Compression: Introduction, coding Redundancy, Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding.

(Text1: Chapters 9, 8, 5)

UNIT – IV	**08 Hours**

Introduction to Computer Vision, Camera Models and Calibration: Camera Projection Models – Orthographic, Affine, Perspective, Projective Geometry, Camera Internal and External Parameters, Lens Distortion Models, Local Feature Detectors and Descriptors: Hessian corner detector, Harris Corner Detector, LOG detector, DOG detector.

(Text3: Chapters 1, 4, 5, 8)(Text4: Chapter 4)

UNIT – V	**07 Hours**
Stereo vision, Epipolar Geometry, Rectification and Issues related to Stereo, SIFT, PCA-SIFT, SURF, HOG, and Image segmentation. (Text3: Chapters 11, 15) (Text4: Chapters 5, 8, 12)	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Comprehend different image representations such as grayscale, RGB and HSV	L2
2	Apply Image Enhancement Techniques including histogram equalization, contrast stretching and sharpening filters	L3
3	Implement spatial domain operations like convolution smoothing filters and edge detection	L4
4	Apply image segmentation techniques like thresholding, region growing and edge-based segmentation	L3
5	Evaluate real-world applications including medical imaging, satellite imagery, face recognition and video processing	L5

Mapping Levels of COs to POs / PSOs

				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
Cos	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO 1	2	2											2	2
CO 2	3	2	2		2								2	1

CO 3	2	2		2									2	2
CO 4	3	2		2	2								2	1
CO 5	3	2	2		2				2				2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Richard Szeliski, "Computer Vision: Algorithms and Applications", Springer, 2nd Edition, 2022.
2. Manas Kamal Bhuyan, "Computer vision and Image Processing Fundamentals and Applications", © 2020 by Taylor & Francis Group.

REFERENCE BOOKS:

1. Tekalp A.M., Digital Video Processing, Prentice Hall (1995).
2. Simon Prince, Computer Vision: Models, Learning, and Inference, 2012.
3. Gonzalez, R.C., and Woods, R.E., Digital Image Processing. 4th edition. Pearson Education (2017).
4. Jain A.K., Fundamentals of Digital Image Processing, Prentice Hall (2015).
5. David Forsyth, Jean Ponce, Computer Vision: A Modern Approach, 2015.

E-Resources:

1. Image Processing/Open CV| Udemy
2. Introduction to Computer Vision and Image Processing | Coursera

Activity-Based Learning (Suggested Activities in Class)

1. The Applications of Computer vision using group discussion.
2. Collaborative Activity is minor project development with a team of 4 students.

****

INNOVATION AND ENTREPRENEURSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	: 24AM3505	**Credits**	: 02
Hours / Week	: 02 Hours	**Total Hours**	: 26 Hours
L–T–P–S	: 2–0–0–0		

Course Learning Objectives:

This course will enable students:

1. Identify and analyze the factors that contribute to the process of successfully launching an entrepreneurial venture and managing a new business.
2. Learn the entrepreneurial process from idea generation to implementation.
3. Acquaint with special problems of starting new ventures, finding products and services, which can support new enterprises, and raising capital.

4. Discuss how to start own business and also to work in or with small business or are involved with entrepreneurship.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

Overview Of Entrepreneurship: The Entrepreneurial Perspective:

Nature and Development of Entrepreneurship. Defining Manager, Entrepreneur, Entrepreneurship and Entrepreneurship. Key Elements of Entrepreneurship. Personality Characteristics of Successful Entrepreneurs. Common Myths about Entrepreneurs. Ethics and Social Responsibility of Entrepreneurs. Types of Start-Up Firms. Process of New Ventu2re7 Creation. Role of Entrepreneurship in Economic Development. Emerging Trends and Issues in Entrepreneurship.

Case Study: Successful Entrepreneurs Narayana Murthy Infosys

UNIT – II	**08 Hours**

THE ENTREPRENEURIAL AND ENTREPRENEURIAL MIND:

The Entrepreneurial Process: Identify and Evaluate the Opportunity, Develop a Business Plan, Determine the Resources Required, Manage the Enterprise. Managerial Versus Entrepreneurial Decision Making: Strategic Orientation, Commitment to Opportunity, Commitment of Resources, Control of Resources, Management Structure, Entrepreneurial Venturing inside a Corporation, Causes for Interest in Entrepreneurship, Climate for Entrepreneurship, Entrepreneurial Leadership Characteristics.

Case study: How to develop effective Business Plan

UNIT – III	**08 Hours**

CREATIVITY AND BUSINESS IDEA:

Identify and Recognizing Opportunities: Observing Trends and Solving Problems. Creativity: Concept,Components and Types of Creativity, Stages of Creative Process. Sources of New Venture Ideas.Techniques for Generating Ideas. Stages of Analyzing and Selecting the Best Ideas. Protecting the Idea:Intellectual Property Rights and its Components. Linking Creativity, Innovation and Entrepreneurship. Case study: Application of Design Thinking in New business ideas generation in particular sector (Health care, Water Saving, Energy saving)

UNIT – IV	08 Hours
Preparing The Proper Ethical And Legal Foundation: Initial Ethical and Legal Issues Facing a New Firm, Establishing a Strong Ethical Culture, Choosing an attorney (Lawyer), Drafting a founder's agreement, Avoiding legal disputes, Choosing a form of business organization, Obtaining business licenses and permits, Choosing a Form of Business Ownership (Sole, Proprietorship, Partnership, Corporation & Limited Liability Company) Case study: Startup Law A to Z IP https://techcrunch.com/2019/02/25/startup-law-a-to-z-intellectual-property/	
UNIT – V	**07 Hours**
Managing Early Growth And Challenges Recruiting and Selecting Key Employees. Lenders and Investors. Funding Requirements: Sources of Personal Financing. Venture Capital. Commercial Banks. Sources of Debt Financing. Key Marketing Issues for New Ventures. Why marketing is critical for Entrepreneurs. Entrepreneurs face unique Marketing Challenges. Guerrilla Marketing. Business Growth: Nature of Business Growth, Planning for Growth, Reasons for Growth. Managing Growth: Knowing and Managing the Stages of Growth, Challenges of Growing a Firm. Strategies for Firms Growth: Internal and External Growth Strategies. Implications of Growth for the Firm and Entrepreneur. Entrepreneurial Skills and Strategies to Overcome Pressures On: Financial Resources (Financial Control, Managing Inventory and Maintaining Good Records). Human Resources, Management of Employees, Time Management. Case study: 9 ways to get startups funded https://www.quicksprout.com/how-to-get-your-startup-funded/	

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Demonstrate knowledge of the key elements of the entrepreneurial process	L2
2	Employ strategies to generate new ideas for startups	L2
3	Outline how to protect IP legally	L2
4	Examine different ways of generating funding	L2
5	Explain organizing managing people, finance and customers	L2

Mapping Levels of COs to POs / PSOs														
Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2

	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2							2		2		2	2
CO2	3	2	2		2				2		2		2	1
CO3	2	2		2					2		2		2	2
CO4	3	2		2	2				2		2		2	1
CO5	3	2	2		2				2		2		2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Dana, Léo-Paul, ed. *World encyclopedia of entrepreneurship*. Edward Elgar Publishing, 2021.
2. Barringer, Bruce R., and R. Duane Ireland Bruce Barringer. *Entrepreneurship: Successfully launching new ventures*. Pearson Education India, 2013.

REFERENCE BOOKS:

1. Soltanifar, Mariusz, Mathew Hughes, and Lutz Göcke. *Digital entrepreneurship: Impact on business and society*. Springer Nature, 2021.
2. Aulet, Bill. *Disciplined Entrepreneurship: 24 Steps to a Successful Startup, Expanded & Updated*. John Wiley & Sons, 2024.
3. Havinal, Veerabhadrappa. Management and entrepreneurship. New Age International, 2009.
4. Janakiram, B. *Management & Entrepreneurship*. Excel Books India, 2010.

E-Resources:

1. https://archive.nptel.ac.in/courses/110/106/110106141/
2. https://www.coursera.org/mastertrack/innovation-management-entrepreneurship-hec

Activity-Based Learning (Suggested Activities in Class)

1. Organizational improvement in startup's using group discussion.

****

CLOUD COMPUTING
(skill Enhancement Course - III)

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Subject Code	:	24AM3506	**Credits**	:	02
Hours / Week	:	03 Hours	**Total Hours**	:	39 Hours
L–T–P–S	:	1–0–2–0			

Course Learning Objectives:

This course will enable students to:

1. **Understand** various basic concepts related to cloud computing technologies.
2. **Contrast** various programming models used in cloud computing.
3. **Choose** appropriate cloud model for a given application.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainmentof the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and role-playing.
3. Show ***Video/animation*** films to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. Discuss how every ***concept can be applied to the real world*** - and when that's possible,it helps improve the student's understanding.

UNIT – I	**08 Hours**

INTRODUCTION TO CLOUD COMPUTING

Introduction, Cloud Computing at a Glance, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model, Characteristics and Benefits, Challenges Ahead, Historical Developments, Distributed Systems, Virtualization, Web 2.0, Service- Oriented Computing, Utility- Oriented Computing

UNIT – II	**08 Hours**

TYPES OF CLOUDS

Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economicsof the Cloud, Open Challenges.

Cloud Definition, Cloud Interoperability and Standards Scalability and Fault ToleranceSecurity, Trust, and Privacy Organizational Aspects.

UNIT – III	**08 Hours**

CLOUD SERVICES AND PLATFORMS

Building Cloud Computing Environments, Application Development, Infrastructure and System Development.

Computing Platforms and Technologies, Amazon Web Services (AWS), Google AppEngine, Microsoft Azure, Hadoop, Force.com and Salesforce.com, Manjrasoft Aneka.

Cloud Computing Architecture-Introduction, Cloud Reference Model- Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service.

Text Book 1: Chapter 1, Chapter 4

UNIT – IV	**08 Hours**

AWS CLOUD APPLICATION PLATFORM

Introduction to AWS ecosystem-Introduction to AWS Elastic Beanstalk-Benefits of using AWS for application development- Supported platforms on Elastic Beanstalk (Java, Python, Node.js, etc.)-Installing AWS CLI & EB CLI-Configuring AWS credentials locally-Creating Elastic Beanstalk environment-Deployment options: CLI, Console, GitHub-Lifecycle of an application in Beanstalk-Modify app configuration-Update code and redeploy changes-Explore environment dashboard-Auto-scaling and load balancing-Monitoring tools: CloudWatch metrics, logs-Health monitoring in Elastic Beanstalk-Connecting Elastic Beanstalk with RDS (Relational Database Service)-Using S3 for static asset storage-IAM roles and security basics

Text Book 1

UNIT – V	**07 Hours**

CLOUD PLATFORMS IN INDUSTRY

Concurrent Computing: Thread Programming, Introducing Parallelism for Single MachineComputation.

Cloud Platforms in Industry, Amazon Web Services, Compute Services, Storage Services, Communication Services, Additional Services, Google AppEngine, Architecture and Core Concepts, Application Life-Cycle, Cost Model, Observations, Microsoft Azure, Azure CoreConcepts, SQL Azure, Windows Azure Platform Appliance.

Text Book 1: Chapter 9

LABORATORY EXPERIMENTS

1. Introduction to Cloud and Virtualization
Objective: Understand cloud computing and install a virtualization environment.

Tools: VirtualBox or VMware

Tasks:

Install VirtualBox

Create and manage a virtual machine (e.g., Ubuntu)

Understand virtualization vs cloud

2. Setting Up a Private Cloud using Eucalyptus/OpenStack (Simulation)
Objective: Simulate a private cloud environment.

Tools: Ubuntu + Eucalyptus (or DevStack for OpenStack)

Tasks:

Install Eucalyptus/DevStack

Launch a VM instance inside the private cloud

Access the dashboard

3. Creating a Virtual Machine in AWS / Azure / GCP
Objective: Deploy a VM in a public cloud.

Tools: AWS EC2 / Azure VM / GCP Compute Engine

Tasks:

Launch an EC2 instance

Connect via SSH

Install a web server (e.g., Apache or Nginx)

4. Deploying a Web Application on AWS Elastic Beanstalk
Objective: Understand PaaS by deploying a web app.

Tools: AWS Elastic Beanstalk, Python/Node.js app

Tasks:

Initialize EB environment

Deploy app using CLI or Console

Monitor deployment logs

5. Using AWS S3 for Object Storage
Objective: Store and retrieve data from cloud storage.

Tools: AWS S3

Tasks:

Create an S3 bucket

Upload and download files

Set permissions and enable static website hosting

6. Load Balancing and Auto Scaling
Objective: Explore scalability in cloud environments.

Tools: AWS Elastic Load Balancer + Auto Scaling Groups

Tasks:

Create multiple EC2 instances

Attach to Load Balancer

Setup auto-scaling rules

7. Monitoring and Logging with CloudWatch
Objective: Monitor performance and logs of cloud resources.

Tools: AWS CloudWatch

Tasks:

Set up basic alarms (CPU usage, memory)

View application logs

Create a dashboard

8. Working with Cloud Databases (RDS / Firebase)
Objective: Use cloud-based relational or NoSQL databases.

Tools: AWS RDS or Firebase

Tasks:

Launch an RDS instance (MySQL/PostgreSQL)

Connect from a web app

Perform CRUD operations

9. Identity and Access Management (IAM)
Objective: Manage user roles and permissions in cloud.

Tools: AWS IAM

Tasks:

Create users and groups

Assign policies (S3 access, EC2 control)

Test access restrictions

10. Final Mini Project – Host a Full-Stack Application on Cloud
Objective: Integrate all concepts in a real-world mini project.

Tasks:

Backend (Node.js/Python) + Frontend (HTML/CSS/JS)

Deployed on EC2 or Beanstalk

Use S3 for static content and RDS for database

Optional Tools & Simulators
CloudSim – For simulating cloud environments in research

Cisco Packet Tracer – For network simulation in cloud labs

Google Cloud Shell – In-browser terminal for experiments

Course Outcomes:

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

1. **Analyze** the principles and components of cloud computing, demonstrating a comprehensive understanding of its architecture and operation.
2. **Compare and contrast** cloud architecture, computing platforms, and technologies.
3. **Analyze** cloud computing architecture principles to design scalable and resilient systems, utilizing various platforms and technologies.
4. **Examine** the framework, cloud deployment mode, and services in Aneka to gain a comprehensive understanding of its functionalities.
5. **Apply** web services and search engines in the industry by utilizing cloud platforms such as Amazon Web Services and Google AppEngine to leverage compute, storage, and communication services.

Cos	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineeringknowledge	Problem analysis	Design	Conduct investigations of complex problems	tool usage	The engineer andsociety	Environment and sustainability	Ethics	teamwork	Communication	Life-long learning	Project managementand finance	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	-	-	-	-	-	-	-	-	-	1		1
CO2	3		-	2	2	-	-	-	-	-	-	1		1
CO3	2	3	2		2	-	-	-	2	-	-	2		1
CO4	3	3	2	-	-	-	-	-	-	-	-	1		1
CO5	2	2	3	-	3	-	-	-	3	-	-	2		1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor**

(Low) TEXT BOOKS:

1. 1. Cloud Computing: From Beginning to End", Ray J. Rafaels Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Packt,2022.
2. Cloud Computing: Principles and Paradigms" by Rajkumar Buyya, James Broberg, and Andrzej Goscinski, Wiley in 2020.

REFERENCE BOOKS:

1. Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, Pearson Education,2020.
2. Cloud Computing A Practical Approach, Anthony T Velte et.al, MC Graw Hill publications, 2014.
3. Cloud Computing Principles and Paradigms, Rajkumar Buyya et.al, WileyPublications, 2015.

Activity Based Learning (Suggestion Activities in Class)

- Presentation
- Collaborative Activity is minor project development with a group of 4 students.

ETHICS, SAFETY & GOVERNANCE OF AGENTIC AI
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER – V

Course Code : 24AM3508 **Credits** : 03

Hours / Week	:	03 Hours	**Total Hours**	:	45(15 hrs for Project)
L–T–P-J	:	2–0–0-2	**CIE** :		100 Marks

Prerequisites:

Basic understanding of Artificial Intelligence, Machine Learning concepts, probability, and introductory programming (Python/C). Familiarity with data ethics and algorithmic decision-making is desirable.

Course Objectives:

This course will enable students to:

1. Understand core ethical theories and principles relevant to the development and operation of agentic and autonomous AI systems.
2. Analyze safety risks, alignment challenges, and potential failure modes in intelligent agents deployed in real-world environments.
3. Evaluate global AI governance frameworks, regulatory policies, and international standards guiding responsible AI adoption.
4. Assess the societal, legal, and economic impacts of agentic AI across critical application domains.
5. Design ethically aligned, safe, and accountable agentic AI solutions incorporating transparency, human oversight, and governance mechanisms.

Teaching-Learning Process (General Instructions)

These pedagogical approaches are designed to accelerate the attainment of course learning outcomes and promote deep conceptual understanding.

1. Lecture Method:

Includes structured lectures integrated with concept-driven discussions supported by contemporary research articles, global policy documents, and real-world case examples of agentic AI systems.

2. Interactive Teaching:

Active learning through guided discussions, ethical dilemma analysis, and problem-solving sessions focused on safety, alignment, and decision-making challenges in autonomous AI agents.

3. Visual Learning:

Use of documentaries, expert talks, system flow visualizations, and simulation videos to illustrate AI failures, governance mechanisms, and human–AI interaction dynamics.

4. Collaborative Learning:

Group-based evaluation of AI governance frameworks, regulatory case studies, and comparative analysis of international AI policies and standards.

5. Critical Thinking:

Faculty will pose higher-order questions encouraging students to reason about ethical trade-offs, accountability, risk mitigation, and long-term implications of agentic AI deployment.

6. Application-Oriented Learning:

Continuous emphasis on applying ethical principles, safety strategies, and governance models to real-world agentic AI use cases across domains such as healthcare, transportation, and public systems.

UNIT – I	**08 Hours**

Foundations of Ethics and Agentic AI

Introduction to Agentic AI and autonomous systems; historical evolution of AI ethics; normative ethical theories (deontology, utilitarianism, virtue ethics); moral agency and responsibility in AI; human–AI interaction and value alignment; overview of ethical challenges in learning, reasoning, and acting agents.

Case Study: Autonomous Decision-Making in Healthcare Triage Systems	
UNIT – II	**08 Hours**
Safety, Risk, and Alignment in Agentic Systems AI safety fundamentals; reward hacking and specification gaming; robustness, reliability, and adversarial risks; alignment problem and value misgeneralization; interpretability and explainability for agentic models; human-in-the-loop and human-on-the-loop systems. **Case Study**: Reward Hacking in a Reinforcement Learning Trading Agent	
UNIT – III	**08 Hours**
Governance, Law, and Policy Frameworks Global AI governance landscape; OECD AI principles; UNESCO AI ethics recommendations; EU AI Act; US AI Bill of Rights; data protection laws (GDPR concepts); accountability, transparency, and auditability; standards bodies (ISO/IEC, IEEE). **Case Study**: Facial Recognition AI and Regulatory Conflict	
UNIT – IV	**08 Hours**
Societal Impact and Responsible Deployment Bias, fairness, and discrimination in agentic AI; impact on employment and labor; AI in healthcare, finance, surveillance, and defense; misinformation and autonomous decision-making risks; sustainability and environmental cost of AI systems; cross-cultural and global south perspectives. **Case Study**: Algorithmic Hiring Agent and Workforce Discrimination	
UNIT – V	**07 Hours**
Future Directions and Trustworthy Agentic AI AI assurance and certification; ethical-by-design and safety-by-design methodologies; red-teaming and impact assessments; governance of autonomous agents and multi-agent systems; future risks (superintelligence, autonomous weapons); research frontiers and global cooperation. **Case Study**: Autonomous Weapons and Human Oversight	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course the student will be able to:		
1	Explain ethical theories and their application to agentic AI systems.	L2
2	Analyze safety risks and alignment challenges in autonomous AI systems	L4
3	Examine global AI governance frameworks, standards, and regulations	L4
4	Assess societal impacts and ethical trade-offs of agentic AI deployment	L3
5	Analyze approaches for responsible and trustworthy governance o agentic AI	L4

Table: Mapping Levels of COs to POs / PSOs														
COs	**Program Outcomes (POs)**												**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	**Problem Analysis**	**Design & Development**	**Conduct Investigations of Complex Problems**	**Modern Tool Usage**	**The Engineer and Society**	**Environment and Sustainability**	**Ethics**	**Individual & Team Work**	**Communication**	**Project management and Finance**	**Life-long Learning**	**Cognitive Outcome**	**Skill & Design Outcome**
CO1	3	2	1	–	–	2	–	–	–	–	–	–	2	2
CO2	3	3	2	2	–	2	–	–	–	–	–	–	2	2
CO3	2	3	2	–	–	3	2	–	–	–	–	–	2	2
CO4	2	2	3	–	–	3	2	–	–	–	–	–	1	1
CO5	3	3	3	2	–	2	–	–	–	–	–	–	1	1

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS (TB):

1. Russell, S., & Norvig, P. *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson, 2021.
2. IEEE Standards Association. *Ethically Aligned Design: A Vision for Prioritizing Human Well-being with Autonomous and Intelligent Systems*. IEEE, 2019.
3. Bostrom, N. *Superintelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.

REFERENCE BOOKS:

1. O'Neil, C. Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy. Crown Publishing Group, 2016.
2. Floridi, L. The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities. Oxford University Press, 2023.
3. AI Now Institute. Annual AI Impact Report. New York University, 2022.
4. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. Concrete Problems in AI Safety. arXiv / OpenAI, 2016.
5. Mittelstadt, B., Allo, P., Taddeo, M., Wachter, S., & Floridi, L. The Ethics of Algorithms: Mapping the Debate. Big Data & Society, 2016.
6. Crawford, K. Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence. Yale

University Press, 2021.

7. European Commission. Ethics Guidelines for Trustworthy AI. Publications Office of the European Union, 2019.
8. OECD. Artificial Intelligence in Society. OECD Publishing, 2019.

E-Resources:

1. https://oecd.ai
2. https://www.unesco.org/en/artificial-intelligence

Activity Based Learning (Suggested Activities in Class)

1. Case study analysis of real-world AI failures.
2. Policy debate role-play (regulator, developer, civil society).
3. Ethical impact assessment of an agentic AI application.
4. Flipped classroom discussions on AI governance documents.

Intelligent Speech Processing Systems [As per Choice Based Credit System (CBCS) scheme] **SEMESTER - V**				
Course Code	**:** 24AM3509	**Credits**	**:**	03
Hours / Week	**:** 03	**Total Hours**	:	45 ((15 hrs for Project)
L–T–P–J	**:** 2–0–0–2	**CIE**	**:**	100 Marks

Course Vision:
This course develops the ability to design, analyse, and evaluate intelligent speech processing and recognition systems by integrating speech signal processing, probabilistic modelling, deep learning, and responsible AI for real-world spoken language applications.

Course Objectives:

Students will:

1. Understand the fundamentals of speech production, perception, and automatic speech recognition pipelines.
2. Apply speech signal processing and feature extraction techniques for speech analysis.
3. Analyze acoustic and language modelling approaches used in conventional ASR systems.
4. Evaluate deep learning and end-to-end methods for modern speech processing applications.
5. Design intelligent speech-enabled solutions with appropriate ethical and engineering judgment.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:
1. Lecture method along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. Interactive Teaching: incorporating brainstorming, discussion, group work, focused listening, problem formulation, note-taking, and classroom demonstrations.
3. Showing video/animation clips and speech demos to explain speech generation, signal analysis, and recognition workflows.
4. Encourage collaborative learning through mini case studies, waveform analysis, and toolkit-based exercises.
5. To make critical thinking, asking higher order thinking questions in the class in the form of quiz, interpretation of speech outputs, and model comparison.
6. Showing different ways to solve the same speech processing problem and encourage the students to come up with creative and responsible solutions.

MODULE – I	**08 Hours**

Foundations of Intelligent Speech Processing
Topics: Introduction to speech processing and speech-based AI systems, speech production mechanism, speech perception, characteristics of speech signals, speech representation, stages of intelligent speech processing systems, and challenges in speech understanding.
Applications: Voice assistants, spoken dialogue systems, dictation systems, and assistive communication tools.

MODULE – II	**09 Hours**

Speech Signal Processing and Feature Extraction **Topics:** Sampling and quantization, pre-emphasis, framing, windowing, short-time analysis, time-frequency representation, spectrogram analysis, feature extraction methods such as MFCC, LPC, PLP, and filter-bank energies. **Applications:** Keyword spotting, speaker interfaces, embedded speech systems, and robust front-end design.	
MODULE – III	**08 Hours**
Acoustic and Language Modelling **Topics:** Acoustic modelling concepts, Hidden Markov Models, Gaussian Mixture Models, Baum-Welch training, Viterbi decoding, pronunciation lexicons, N-gram language models, and decoding strategies in conventional ASR systems. **Applications:** IVR systems, command-and-control interfaces, transcription engines, and speech analytics pipelines.	
MODULE – IV	**10 Hours**
Deep Learning for Speech Processing **Topics:** Limitations of traditional ASR, deep neural networks for speech, Transformer concepts in speech processing, self-supervised learning, end-to-end ASR, CTC, and attention-based speech models. **Applications:** Real-time transcription, multilingual speech interfaces, mobile ASR, and cloud speech services.	
MODULE – V	**10 Hours**
Evaluation, Multilingual Speech Applications, and Emerging Trends **Topics:** Speaker-dependent and speaker-independent systems, noise-robust speech recognition, multilingual and low-resource speech processing, evaluation metrics such as WER, SER, and CER, speech APIs and toolkits, ethics, privacy, fairness, and recent research trends. **Applications:** Healthcare transcription, customer support automation, educational technology, accessibility systems, and domain-specific spoken language solutions.	

AI Integration:

- AI tools for waveform, spectrogram, and speech feature understanding; AI-assisted explanation of speech pipelines.
- Classical ASR modelling support using AI for comparing HMM, GMM, lexicon, and language model workflows.
- Deep learning based speech models, speech embeddings, and transformer-based ASR prototyping tools.
- Pretrained speech APIs, toolkit benchmarking environments, and multilingual speech processing platforms.
- Evaluation of intelligent speech systems using standard datasets, recognition metrics, and responsible AI validation.

Authentic Intelligence:

- Limitations of machine speech perception compared with human auditory perception and context understanding.
- Bias due to accents, languages, dialects, and data imbalance in speech datasets.

- Challenges in noisy speech, multilingual speech processing, and over-reliance on AI-generated modelling suggestions.
- Ethical concerns in speech technologies including privacy, inclusion, fairness, transparency, and deployment risks.
- Students evaluate societal impact, data sensitivity, and practical trustworthiness of intelligent speech systems.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the fundamentals of speech production, perception, and speech signal characteristics.	L2
2	Apply speech signal processing and feature extraction techniques for speech analysis.	L3
3	Analyze acoustic and language modelling approaches used in conventional ASR systems.	L4
4	Evaluate deep learning and end-to-end speech processing models for recognition tasks.	L5
5	Design intelligent speech-enabled applications for multilingual and real-world environments.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	1	–	1	–	–	–	–	–	1	1	2
CO-2	3	3	2	2	2	–	2	2	–	1	1	2	3
CO-3	3	3	3	2	2	–	1	1	–	1	1	2	3
CO-4	3	2	3	3	3	–	2	2	–	2	2	3	3
CO-5	3	3	3	3	3	1	1	–	–	2	2	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H., Speech and Language Processing, 3rd Edition, Pearson,
2. Rabiner, L. R., & Juang, B.-H., Fundamentals of Speech Recognition, Pearson, year..
3. Quatieri, T. F., Discrete-Time Speech Signal Processing, 3rd Edition, Pearson.

REFERENCE BOOKS:

1. Yu, D., & Deng, L., Automatic Speech Recognition: A Deep Learning Approach, Springer.
2. Deng, L., & Yu, D., Deep Learning for Speech Recognition and Language Processing, Springer.
3. O'Shaughnessy, D., Speech Communications: Human and Machine, 2nd Edition, Wiley-IEEE Press.

E-RESOURCES:

1. Research papers from IEEE, Springer, Elsevier, and ACL Anthology related to speech processing and ASR.
2. NPTEL / SWAYAM courses on Speech Processing, Pattern Recognition, and Machine Learning.
3. Official documentation of Librosa, Kaldi, SpeechBrain, torchaudio, and Hugging Face speech models.

RECOMMENDED TOOLS

- Programming: Python, Jupyter Notebook
- Libraries: Librosa, torchaudio, SpeechBrain, Kaldi, Hugging Face Transformers
- Data Processing: NumPy, SciPy, pandas
- Visualization: Matplotlib, waveform and spectrogram analysis tools
- AI Tools: ChatGPT, code assistants, speech API sandboxes, and model benchmarking platforms

FOUNDATIONS OF ROBOTICS AND INTELLIGENT SYSTEMS

SEMESTER - V

Subject Code	:	24AM3510	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–J :	:	2–0–0–2	**CIE**	:	100

Course Vision:

To establish a strong foundational understanding of robotic systems by integrating modeling, kinematics, motion representation, and sensing principles, enabling students to systematically analyze and interpret robotic behavior. The course aims to build the essential theoretical and conceptual base required for advanced topics such as autonomous systems, robot intelligence, and human–robot interaction, forming a critical bridge between fundamental engineering concepts and modern robotics applications.

Course Learning Objectives:

Students will:

1. Understand the fundamental components and architecture of robotic systems.
2. Model robotic motion using coordinate transformations and kinematic representations.
3. Analyze basic trajectory generation and motion characteristics in robotic systems.
4. Understand sensing principles and their role in robotic perception.
5. Interpret the role of actuation and control in enabling robotic motion.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe robotic system components and classifications	L4
CO2	Apply coordinate transformations and kinematic	L3
	models for robotic manipulators	
CO3	Analyze motion representation and basic trajectory	L4
	planning in robotic systems.	
CO4	Evaluate sensing mechanisms and their role in robotic	L5
	perception.	
CO5	Examine basic robotic systems for simple applications.	L4

UNIT – I : Introduction to Robotic Systems **09 Hours**

Robot definition, automation vs robotics, robot classification, robot anatomy, degrees of freedom, workspace, introduction to coordinate frames and reference systems.
Applications: Industrial robots, service robots, mobile robots

UNIT – II : Robot Kinematics and Motion Representation	**12 Hours**
Coordinate transformations, rotation matrices, homogeneous transformations, coordinate frames, Denavit-Hartenberg (D-H) parameters (conceptual), forward and inverse kinematics (conceptual), representation of motion in joint space and task space, introduction to Jacobian (conceptual), velocity mapping between joint space and task space, basic understanding of singularities. Applications: Robotic manipulators, medical robotics	
UNIT – III : Motion and Actuation in Robotic Systems	**09 Hours**
Basic trajectory planning concepts, types of robotic motion (point-to-point and continuous), joint space vs task space trajectories, overview of actuators (electric, pneumatic, hydraulic), DC motors, servo motors, stepper motors (functional understanding). Applications: Pick-and-place systems, robotic assembly	
UNIT – IV : Sensing and Introduction to Robotic Perception	**09 Hours**
Position and velocity sensing, encoders, tactile sensors, proximity sensors, introduction to perception in robotics, basic sensor data interpretation, role of sensors in enabling environment interaction. Applications: Smart manufacturing, autonomous systems, inspection robots	
UNIT – IV : Application, Case Studies, and System-Level Modeling	**6 Hours**
Integration of robotic system components including kinematics, motion, actuation, and sensing; system-level understanding of robotic workflows; case studies of robotic systems (industrial manipulators, mobile robots, service robots); modeling and simulation of simple robotic systems; interpretation of robot behavior through system representation; introduction to basic simulation tools for robotic systems. Applications: Pick-and-place robots, mobile robot navigation (conceptual), service robotics systems	

Mapping Levels of COs to POs / PSOs													
				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and World	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	-	1	-	-	-	-	-	1	3	1
CO2	3	3	2	1	1	-	-	-	-	-	2	3	2
CO3	3	2	2	1	1	-	-	1	-	-	2	3	2
CO4	2	2	1	1	1	-	1	1	-	-	1	2	1
CO5	3	2	3	1	2	-	-	2	1	1	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

Role of sensing and data in intelligent robotics

Learning-based approaches for motion approximation (overview Adaptive motion and trajectory optimization (conceptual)

Sensor data processing and basic sensor fusion concepts

Authentic Intelligence Reflection:

Human supervision and limitations of automated systems
Reliability and limitations in robotic motion modeling

Safety considerations in robot motion and human interaction

Ethical considerations and reliability of sensing systems

Mini Project (Mandatory)

Students will design and simulate a basic robotic system using modeling and representation concepts.

Example Problem Statements:

- Simulation of a robotic manipulator using forward kinematics
- Basic trajectory generation for a robotic arm
- Modeling and visualization of robot motion using coordinate transformations
- Sensor-based simple robotic system simulation

Students must demonstrate:

- System modeling and representation

- Motion analysis and visualization
- Basic simulation and validation

Evaluation Focus:

- Correctness of modeling
- Clarity of system representation
- Simulation quality
- Understanding of robotic motion principles

Recommended Tools

- MATLAB Robotics Toolbox
- Python (NumPy, basic simulation libraries)
- Arduino (optional for simple demonstrations)
- Basic ROS visualization (optional)

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

Text Books

1. Saeed B. Niku, Introduction to Robotics: Analysis, Control, Applications, 3rd Edition, John Wiley & Sons, 2020.
2. Groover, Mikell P., Industrial Robotics: Technology, Programming and Applications, 1st Edition, McGraw-Hill Education, 2013.

Reference Books

1. Yan, Lili & Gene M. Grossman, Robots and AI: A New Economic Era, 1st Edition, Taylor & Francis, 2023.
2. Klafter, Richard D., Thomas A. Chmielewski & Michael Negin, Robotic Engineering: An Integrated Approach, 1st Edition, Prentice Hall, 2013.
3. Fu, K. S., R. C. Gonzalez & C. S. G. Lee, Robotics: Control, Sensing, Vision and Intelligence, 1st Edition, McGraw-Hill, 2013.

****

Explainable AI

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - V

Course Code	: 24AM3511	**Credits**	: 03
Hours / Week	: 03	**Total Hours**	: 45 (15 hrs for Project)
L–T–P–J	: 2–0–0–2	**CIE**	: 100 Marks

Course Vision:

To enable learners to develop transparent, interpretable, and trustworthy AI systems by understanding and applying Explainable AI techniques in real-world engineering applications.

Course Objectives:

This course will enable students to:

1. To understand the fundamental concepts, need, and challenges of Explainable AI.
2. To explore various taxonomy of XAI methods, including model-agnostic and model-specific approaches.
3. To study and apply interpretable machine learning models for transparent decision-making.
4. To analyze explainability techniques in deep learning models using visualization and attribution methods.
5. To implement and evaluate XAI techniques using tools and real-world case studies.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. Interactive Teaching: incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing Video/animation films to explain functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, asking Higher order Thinking questions in the class in the class in the form of Quiz and writing programs with complex solutions.
6. Showing different ways to solve the same problem and encouraging the students to come up with their own creative ways to solve them

MODULE – I :	**09 Hours**

Black Box Problem in AI, History of AI & Explainability, Need for Explainability: Trust, Transparency, Accountability, Definitions, Goals, and Scope of XAI, Science of Interpretable Machine Learning, Motivation, Challenges, and Trade-offs of Explainable AI, Human Factors in Explainability, Interpreting Interpretability.

Engineering Applications: Healthcare diagnosis systems, transparent credit scoring in financial systems, AI-based recommendation systems with user trust

MODULE – II	**09 Hours**
Model-Agnostic Approaches: LIME, SHAP, Integrated Gradients. Model-Specific Approaches: Rule-based Explanation, Tree Feature Importance. What-if Analysis: Counterfactual Explanations, Reasoning-based: Causal Inference and User-facing: Interactive Explanations and Visualizations. Engineering Applications: Explainable Credit Scoring System (Finance), Healthcare Diagnosis Support System,Counterfactual Explanation in Insurance Systems	
MODULE – III	**09 Hours**
Interpretable vs. Explainable Algorithms, Ensemble-Based-Boosted Rulesets, Explainable, Boosting Machines (EBM), RuleFit, Skope-Rules, Iterative Random Forests (iRF), Decision Tree, Based-Optimal Classification Trees, Voting, stacking, Optimal Decision Trees, Scoring System Engineering Applications: High-stakes decision systems like healthcare, finance, and risk assessment	
MODULE – IV	**09 Hours**
Applications, Tools and Libraries, Intrinsic, Perturbation- LIME, Occlusion, RISE, Prediction, Difference Analysis, Meaningful Perturbation, Gradient/Backpropagation Activation, Maximization, Class Model Visualization, Saliency Maps, DeepLIFT, DeepSHAP, Deconvolution, Guided Backpropagation, Integrated Gradients, Layer-Wise Relevance Propagation, Excitation, Backpropagation, CAM. **Engineering Applications: Computer vision, medical imaging, and autonomous systems**	
MODULE – V	**09 Hours**
Hands-on implementation of XAI libraries (e.g., LIME, SHAP). Explainability metrics and evaluation methods. Trade-offs between model performance and explainability. Case studies of XAI applications in real-world scenarios. Engineering Applications: Applied in real-world AI deployment and model validation systems	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the importance of interpretability and transparency in AI systems.	L2
2	Differentiate between various XAI methods such as model-agnostic, model-specific, and causal approaches.	L4
3	Apply interpretable machine learning models for solving real-world problems	L3
4	Analyze and interpret deep learning models using advanced XAI techniques	L4
5	Evaluate and implement explainable AI solutions considering performance and interpretability trade-offs.	L5

Mapping Levels of COs to POs / PSOs

Mapping Levels of COs to POs / PSOs													
				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Cognitive Outcome	Skill & Design Outcome
CO1	2	2	-	-	1	-	-	2	-	-	-	1	2
CO2	2	3	-	2	1	-	-	2	-	-	-	2	2
CO3	3	3	2	2	2	-	-	1	-	-	-	3	3
CO4	3	3	2	3	2	-	-	1	-	-	-	3	3
CO5	3	3	3	3	2	1	1	3	1	1	1	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**

1. Understanding transparency and interpretability in AI-driven decision systems.
2. Application of model-agnostic and model-specific XAI techniques for explaining machine learning models.
3. Use of interpretable machine learning models for transparent prediction and decision-making.

4. Explainability techniques for deep learning models using visualization and attribution methods.
5. Evaluation, deployment, and ethical integration of explainable AI in real-world systems.

- **Authentic Intelligence Reflection:**

1. Reflect on the importance of transparency and trust in AI systems and their impact on human decision-making.
2. Analyze how different explanation techniques influence understanding, fairness, and accountability of AI models.
3. Evaluate the balance between model accuracy and interpretability in real-world problem-solving scenarios.
4. Interpret deep learning model behaviors using visualization techniques and reflect on their reliability
5. Assess ethical implications, biases, and societal impact of deploying explainable AI systems.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Samek, W., Montavon, G., Vedaldi, A., Hansen, L. K., & Müller, K.-R. (2019). Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Springer.

2. Molnar, C. (2020). Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. Independently published.

REFERENCE BOOKS:

1. Raschka, S., & Mirjalili, V. (2019). Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2 (3rd ed.). Packt Publishing.
2. Mayuri Mehta, Vasile Palade, Indranath Chatterjee, "Explainable AI: Foundations, Methodologies and Applications", Springer, 2023.

E-RESOURCES:

1. https://nvdam.widen.net/s/wlbgbqr7cj/nvidia-learning-training-course-catalog
2. https://www.coursera.org/learn/explainable-ai-xai

RECOMMENDED TOOLS

- Python, OpenCV
- TensorFlow / PyTorch
- Scikit-learn
- Jupyter Notebook

COGNITIVE LINGUISTICS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – V

Course Code	: 24AM3512	**Credits**	: 03
Hours / Week	: 03 Hours	**Total Hours**	: 45 (15 hrs for Project)
L–T–P-J	: 2–0–0-2	**CIE:** 100 Marks	

Course Vision

Course Objectives:

1. To introduce students to the historical development, core assumptions, and main schools of thought within cognitive linguistics as an alternative to formal models of language.
2. To develop students' understanding of key theoretical frameworks in cognitive linguistics, including frame semantics, construction grammar, conceptual metaphor theory, and prototype theory.
3. To enable students to analyse linguistic data using usage-based, functional, and cognitive approaches, linking grammar to cognition, categorization, and language use.
4. To train students to critically read, interpret, and discuss primary scholarly works in cognitive linguistics, fostering independent thinking and methodological awareness.
5. To cultivate students' ability to design and present small-scale research projects in cognitive linguistics, culminating in oral presentations and a written term paper.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only traditional lecture methods, but different *types of teaching methods* may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt the* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and roleplaying.
3. Show ***Video/Animation*** films to explain functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher order Thinking questions in the class.

6. Adopt ***Problem Based Learning***, which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyse information rather than simply recall it.
7. Show the ***different ways to solve*** the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the students' understanding.

MODULE– I	**09 Hours**

INTRODUCTION AND ORIGINS OF COGNITIVE LINGUISTICS: Cognitive linguistics and other cognitive sciences.
Formal, functional and cognitive linguistics; historical development and core assumptions.

MODULE – II	**09 Hours**

Grammar, Cognition and Usage-Based Model: Grammar and cognition; Talmy's cognitive representations.
Language as human behaviour, language acquisition, language as a complex adaptive system (usage-based model).

MODULE – III	**09 Hours**

Frame Semantics and Construction Grammar: Frame semantics: frames and Fillmore's case grammar.
Construction grammar: constructions as elements of grammar; Fillmore, Langacker; construction grammar vs cognitive grammar.

MODULE – IV	**09 Hours**

Conceptual Metaphor and Conceptual Blending: Lakoff & Johnson: metaphor as a cognitive mechanism.
Fauconnier: conceptual blending and its applications in meaning construction.

MODULE – V	**09 Hours**

Prototype Theory, Typology and Applications: Prototype theory: Rosch, Berlin & Kay;

prototype and basic-level categories; applications in psychology, psychiatry, cultural anthropology. Typology and cognitive linguistics: Corbett, language universals, future directions; student project presentations as integrative activity.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand the historical development and core assumptions of cognitive linguistics and distinguish it from formal and functional approaches to language.	**L2**
2	Apply concepts of grammar–cognition interfaces and usage-based models to analyse how language structure emerges from use and cognitive processes.	**L3**
3	Analyze linguistic data using frame semantics and construction grammar to identify how meaning and grammatical patterns are organised in speakers' minds.	**L4**
4	Analyze conceptual metaphors and conceptual blending in authentic language examples to reveal underlying cognitive mechanisms of meaning construction.	**L4**
5	Evaluate and design small-scale studies that use prototype theory and typological perspectives to explain categorization and linguistic universals in cognitive linguistics.	**5**

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)												PSOs	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	2	1	0	0	0	1	0	1	0	1	0	2	1	1

CO2	2	2	1	1	1	1	0	1	0	1	0	2	1	2
CO3	2	3	1	2	1	1	0	1	1	2	1	2	2	2
CO4	1	2	1	2	1	2	1	2	1	3	1	2	2	2
CO5	2	3	2	3	2	2	1	2	2	3	2	3	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

- CO3 and CO5 have higher mapping to PO2 and PO4 because students analyse data and synthesize conclusions from research-like tasks and term projects.
- CO4 and CO5 map more strongly to PO10 and PO12 due to presentations, term papers, and emphasis on continuous engagement with scholarly literature.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS (TB):

1. Evans, Vyvyan, & Green, Melanie. Cognitive Linguistics.
   - A comprehensive, up-to-date general introduction to cognitive linguistics covering cognitive semantics, construction grammar, metaphor, blending, and comparisons with formal approaches.
   - Intended as a primary textbook for undergraduate and postgraduate courses, with clear structure aligning well with units on frames, constructions, metaphor, and typology.
2. Dąbrowska, Ewa, & Divjak, Dagmar (eds.). Handbook of Cognitive Linguistics. De Gruyter, 2015.
   - State-of-the-art handbook with chapters on usage-based grammar, frame semantics, construction grammar, metaphor, prototypes, and typology by leading scholars.researchportal.northumbria+1
   - Suitable as an advanced text to support student projects and term papers, giving research-level depth beyond the core textbook.
3. Dąbrowska, Ewa (ed.). The Cambridge Handbook of Cognitive Linguistics. Cambridge University Press, 2017.
   - Provides a thorough overview of methods, key concepts, and new subfields in cognitive linguistics, including construction grammar, conceptual mappings, acquisition, corpus studies, and diachrony.
   - Well-suited as a reference and complementary reading for units on grammar–cognition, metaphor/blending, and prototype theory.

REFERENCE BOOKS:

1. **Tomasello, Michael. A Natural History of Human Thinking. Harvard University Press, 2014 (often used post-2015).**
    - Explores how uniquely human forms of thinking and shared intentionality underpin language and communication, complementing usage-based and cognitive views of grammar and meaning.
2. **Bybee, Joan. Frequency of Use and the Organization of Language. Oxford University Press, 2006 (still a key reference in 2015–2025 syllabi).**
    - Classic usage-based account of how frequency and experience shape grammar, morphology, and constructions, directly supporting units on usage-based models and construction grammar.

.

E-Resources:

NPTEL courses

- **Language, Culture and Cognition: An Introduction (IIT Guwahati)** – Introduces relations between language, cognition and culture explicitly from a cognitive linguistics perspective; videos and materials are free, exam is paid only if needed.
- **Bilingualism: A Cognitive and Psycholinguistic Perspective (NPTEL)** – Explores cognitive and psycholinguistic aspects of bilingualism, useful for units on cognition and categorisation.onlinecourses.
- **Applied Linguistics (NPTEL)** – Focus on applications of linguistic theory; relevant for connecting cognitive and applied perspectives.
- **Introduction to Cognitive Psychology / Cognition and its Computation / Introduction to Basic Cognitive Processes (NPTEL)** – Background on cognitive processes, attention, memory and computation supporting cognitive linguistics foundations.

audit-mode MOOCs

- **Coursera – Linguistics & Cognitive Science courses** – Several linguistics and cognitive science courses can be audited free (access to videos and readings; certificate is paid).
- **FutureLearn – Free Linguistics Courses** – Short, fully online linguistics courses (often free for the duration of the course) that cover language, communication and sometimes language & mind topics.
- **Alison – Cognitive Processing and Language Acquisition** – Free course on cognitive processing and language acquisition, useful as a supplementary psycholinguistics resource.

Activity Based Learning (Suggested Activities in Class)

1. Real world problem solving using group discussion.
2. Role play E.g., Stack, Queue, etc.,
3. Demonstration of solution to a problem through programming.

Flip class activity E.g., arrays, pointers, dynamic memory allocation, etc.,

VI SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	IPCC	24AM3601	Deep Learning	3	0	2	0	60	40	100	04
2	IPCC	24AM3602	Generative AI	3	0	2	0	60	40	100	04
3	PCC	24AM3603	Designing MLOps for Enterprises	2	0	2	0	60	40	100	03
4	PEC	24AM36XX	Professional Elective Course – II	2	0	0	2	100	--	100	03
5	PEC	24AM36XX	Professional Elective Course – III	2	0	0	2	100	--	100	03
6	OEC	24OEXXXX	Open Elective – I	3	0	0	0	60	40	100	03
7	PROJ	24AM3604	Minor Project	0	0	0	4	--	100	100	02
8	AEC	24AM3605	Cognitive and Technical Skills-VI	-	-	-	-	--	--	--	P/F
			Total	15	0	6	8				**22**

IPCC-Integrated Professional Core Course, AEC-Ability Enhancement Course,PCC-Professional Core Courses, PEC-Professional Elective Courses, OEC-Open Elective Courses, PROJ-Project Work, L – Lecture, T – Tutorial, P – Practical, J– Project, C – No. Of Credits

Professional Elective Courses –II (PEC-II) Offering										
Sl. No	**Course Type**	**Course Name**	**Lecture**	**Tutoria**	**Practic**	**Project**	**CIE Marks**	**SEE Marks**	**Total Marks**	C
			L	**T**	**P**	**J**				
1	PEC	LLM based Agent Frameworks	2	0	0	2	100	--	100	**03**
2	PEC	Visual Computing and Graphics	2	0	0	2	100	--	100	**03**
3	PEC	Robot Operating Systems (ROS)	2	0	0	2	100	--	100	**03**
4	PEC	AI for Autonomous Vehicles & Smart Mobility	2	0	0	2	100	--	100	**03**
5	PEC	Conversational AI and spoken Language Understanding	2	0	0	2	100	--	100	**03**
DLI INTEGRATED										
6	DLI/MOOC	Deep Tech systems								

Professional Elective Courses –III (PEC-III) Offering										
Sl. No	**Course Type**	**Course Name**	**Lectur**	**Tutori**	**Practi**	**Projec**	**CIE Marks**	**SEE Marks**	**Total Marks**	C
			L	**T**	**P**	**J**				
1	PEC	Multi-Agent AI Systems	2	0	0	2	100	--	100	**03**
2	PEC	Intelligent Computational Imaging and Reconstruction	2	0	0	2	100	--	100	**03**
3	PEC	Autonomous Robotic Systems	2	0	0	2	100	--	100	**03**
4	PEC	AI for Healthcare	2	0	0	2	100	--	100	**03**
5	PEC	Building AI Applications with Large Language Models	2	0	0	2	100	--	100	**03**
DLI INTEGRATED										
5	DLI/MOOC	Graphics and Simulation								

Open Elective-1 (OE-1) Offering										
Sl. No	**Course Type**	**Course Name**	**Lectu**	**Tutor**	**Practi**	**Proje**	**CIE Marks**	**SEE Marks**	**Total Marks**	
			L	**T**	**P**	**J**				
1	OEC	Industrial Robotics	3	0	0	0	60	40	100	**03**
2	OEC	Machine Learning for Health Care	3	0	0	0	60	40	100	**03**
DLI INTEGRATED										
3	DLI	Accelerated Computing								

DEEP LEARNING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3601	**Credits**:	04
Hours / Week	: 05 Hours	**Total Hours**:	39 (T) +26(p)
L–T–P–S	: 3–0–2–0		

Course Learning Objectives:

This course will enable students to:

1. To understand the basic building blocks and general principles that allow oneto design Deep learning algorithms
2. To become familiar with specific, widely used Deep learning networks
3. To introduce building blocks of Convolution neural network architecture
4. To learn to use deep learning tools and frameworks for solving real-life problems

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** clips to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

MODULE 1: INTRODUCTION TO DEEP LEARNING **08 Hours**

Introduction to Perceptron, Learning rules, Single layer perceptron and its applications, Multilayer Feed forward Neural Network, neural networks training, activation functions, loss functions, Model Selection.

(Text 2: Chapters 2&3)

MODULE 2: COMPUTATION IN DEEP LEARNING **07 Hours**

Forward and Backward Propagation, Computational Graphs Layers, shallow neural network, deep neural network, Optimization for Training Deep Models, Gradient Descent and its types, Regularization techniques, Automatic Differentiation. (Text 1: Chapter 4)

MODULE 3: CONVOLUTIONAL NEURAL NETWORKS **08 Hours**

Convolutional Neural Networks (CNNs) - Biological inspiration, Mapping of Human Visual System and CNN. Convolution operation, Convolutional Layers, Padding and Stride, Subsampling - Pooling, Fully Connected CNN Layer, Transfer Learning, Overview of Alex Net, VGG, Google Net, Res Net, YOLO, Graph Neural Networks.

(Text1: Chapter 7)

MODULE 4: SEQUENCES TO MODEL **08 Hours**

Basics of RNN, Types of RNN, RNN's For Sequence Modeling- Language Modeling, Back Propagation Through Time, LSTM Network, Bidirectional LSTMs, Applications of LSTM.
(Text 1: Chapter 9, 10)

MODULE 5: UNSUPERVISED DEEP LEARNING **08 Hours**

Unsupervised Pretrained Networks (UPNs) - Autoencoders, Deep Belief Networks (DBNs), Generative Adversarial Networks (GANs), Restrictive Boltzmann Machines, Transformers.
Certification/Case Study: Applications of AI for Anomaly Detection, Applications of AI for Predictive Maintenance, Bootstrapping Computer Vision Models With Synthetic Data.
Text 1: chapter 20

LAB Experiments

1. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets. Vary the activation functions used and compare the results.
2. **Design** and build a comprehensive project leveraging various deep learning architectures to solve real-world problems effectively
3. Analyze the performance of CNN on the image dataset.
4. Load the dataset as input.
5. Change the hyper-parameter (Varying with different convolutional and pooling layer) of classification model and analyze its performance.
6. Evaluate and compare the model performance by using metrics – classification accuracy and Binary Cross Entropy Loss.
7. Design and implement a CNN model (with 2+ layers of convolutions) to classify multi category image datasets. Use the concept of padding and Batch Normalization while designing the CNN model. Record the accuracy corresponding to the number of epochs. Use the Fashion MNIST/MNIST/CIFAR10 datasets.
8. Design and implement a CNN model (with 4+ layers of convolutions) to classify multi category image datasets. Use the concept of regularization and dropout while designing the CNN model. Use the Fashion MNIST/MNIST/CIFAR10 datasets datasets. Record the Training accuracy and Test accuracy corresponding to the following architectures:

   I. Base Model
   II. Model with L1 Regularization
   III. Model with L2 Regularization
   IV. Model with Dropou
   V. Model with both L2 (or L1) and Dropout

9. Implement the standard VGG-16, Resnet architecture model to classify multi category image dataset and check the accuracy.
10. Implement the standard Densenet architecture model to classify multi category image dataset and check the accuracy.
11. Implement and Compare RNN and LSTM for sentiment analysis on movie reviews/other dataset and tabulate the results.
12. Implement and Compare LSTM and Bidirectional LSTM for sentiment analysis on movie reviews/other dataset and tabulate the results.

13. Implement Generative Adversarial Networks to generate realistic Images. Use MNIST, Fashion MNIST or any human face datasets.

14. Implement Auto encoders for image denoising on MNIST, Fashion MNIST or any suitable dataset.

Open Ended Experiments:

1) "Smart Diagnosis" – Disease Prediction Using Medical Imaging.
2) "Smart Yield" – Predicting Crop Yield with Satellite and Environmental Data.
3) "Stock Sense" – Predicting Stock Market Trends Using Deep Learning.

Course Outcomes:

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

1. **Apply** deep learning models such as Artificial Neural Networks (ANN) and Convolutional Neural Networks (CNN) for classification tasks on various datasets.
2. **Evaluate and compare** the performance of deep learning architectures using different hyperparameters and regularization techniques on image data.
3. **Implement** Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks for learning, predicting, and classifying real-world problems within the deep learning domain.
4. **Apply** Autoencoders and Generative Adversarial Networks (GANs) for tasks such as image denoising and data augmentation.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Build an Image classifier model for applying the concept of single layer and multilayer NN and analyze activation and loss function with that model.	L2
2	Apply the mathematical concept of deep learning for the manipulation and preprocessing of data.	L3
3	Evaluate deep learning models applying optimization techniques to solve real-world problems and analyse the efficiency of the models.	L5
4	Build an image classifier model, applying CNN and evaluating associated hyperparameters.	L4, L5
5	Construct deep learning-based models for healthcare applications and compare effectivity of advanced networks.	L5

Mapping Levels of COs to POs / PSOs														
COs				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	3	-	2	-	-	-	2	-	2	-	1	1
CO2	3	3	3	3	2	-	-	-	2	-	2	-	2	2
CO3	3	3	3	3	2	-	-	-	2	-	2	-	2	2
CO4	3	3	3	3	2	-	-	-	2	-	2	-	3	3
CO5	3	3	3	3	2	-	-	-	2	-	2	-	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Aston Zhang, Zack C. Lipton, Mu Li, Alex J. Smola, "Dive into Deep Learning", Amazon Science, 2020
2. Josh Patterson and Adan Gibson, "Deep Learning a Practitioners Approach", July 2018.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", The MIT Press, 2016

REFERENCE BOOKS:

1. Tom Mitchell, Machine Learning, McGraw-Hill, 1997
2. François Chollet, "Deep Learning Python", Manning Publications, 2018
3. Neural Networks: A Comprehensive Foundation," S. Haykin, 2ndEd, Prentice Hall of India, 2003.

Activity-Based Learning (Suggested Activities in Class)

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity)
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems about the syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

****

GENERATIVE AI
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER - VI

Course Code	**:**	24AM3602	**Credits**	**:**	04
Hours / Week	**:**	05	**Total Hours**	**:**	45+30
L–T–P–J	**:**	3–0–2–0	**CIE+SEE**	**:**	60+40 Marks

Course Vision:
To develop students' ability to understand, design, and apply generative AI models across text, image, and multimodal domains, while integrating AI-assisted tools, critical evaluation of generated outputs, NVIDIA DLI certified learning pathways, and responsible engineering judgment.

Course Objectives:
This course will enable students to:

1. To understand the fundamental concepts, architectures, and applications of Generative AI across modalities.
2. To apply prompt engineering techniques for effective interaction with generative models and large language models.
3. To analyze generative model architectures including VAEs, GANs, transformers, GPT, and diffusion models.
4. To evaluate and compare different large language models for real-world text processing applications.
5. To design AI agent workflows and autonomous systems with planning, memory, and tool use capabilities.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Introduction to Generative AI

Topics: Landscape and hierarchy of Generative AI (conceptual framing pre-studied via DLI); domain-specific applications in healthcare, finance, creative industries, engineering; hands-on categorization of GenAI tools by architecture and modality (ChatGPT, Claude, DALL-E, Stable Diffusion); ethical considerations — bias amplification, deepfakes, intellectual property, environmental cost of training.

Applications: Content generation in media, drug discovery, synthetic data for engineering simulations, automated design and prototyping.

MODULE – II	**09 Hours**

Prompt Engineering **Topics:** Synthesis and critique of prompt engineering techniques: classroom discussion focuses on when prompt engineering reaches its limits versus when fine-tuning or architectural changes are needed; defensive prompting, prompt injection, jailbreaking awareness; security and safety considerations not covered in the DLI course. **Applications:** Automated report generation, code generation assistants, customer service automation, educational content creation.	
MODULE – III	**09 Hours**
Generative AI Architectures — Transformers, GPT, Diffusion **Topics:** Encoder/decoder architectures; latent space in generative models; transformer architectures and self-attention mechanism; multi-head attention and positional encoding; conditional generative models; GPT architecture (pre-training, fine-tuning, inference); encoder-only (BERT), decoder-only (GPT), encoder-decoder (T5) comparison; denoising diffusion models for text-to-image pipelines; introduction to retrieval-augmented generation (brief, complemented by DLI). **Applications:** Machine translation, text summarization, code completion, image captioning, question answering, text-to-image generation.	
MODULE – IV	**09 Hours**
Language Models & LLM Architectures **Topics:** Language models and tokenization fundamentals; LLM training and inference at a conceptual level; LLMs vs traditional Generative AI; examples of leading LLMs (GPT-4, Claude, LLaMA, Gemini); fine-tuning vs prompt engineering vs RAG — decision framework; model evaluation metrics (perplexity, BLEU, ROUGE, human evaluation); inference sizing and optimization (covered in depth via DLI). **Applications:** Sentiment analysis, document classification, chatbots, knowledge extraction, legal document analysis, scientific literature review.	
MODULE – V	**09 Hours**
AI Agents & Agentic Workflows **Topics:** AI agents and autonomous systems overview; core components of intelligent agents — Planning, Memory, Tool Use, Goal-directed behavior; agent orchestration frameworks (LangChain, LangGraph, CrewAI, AutoGen); RAG agents; Chain-of-thought multi-step reasoning; NVIDIA NIM for agent deployment (hands-on via DLI); agent deployment strategies and monitoring. **Applications:** Autonomous research assistants, multi-agent software development, enterprise workflow automation, supply chain optimization.	

Authentic Intelligence :

1. Module 1 — Introduction to GenAI: Students critically assess hallucination, bias amplification, and the environmental cost of training; they evaluate ethical considerations around deepfakes and intellectual property and reflect on when AI-generated content requires mandatory human verification.

2. Module 2 — Prompt Engineering: Students evaluate when prompt engineering is the right tool versus when fine-tuning or RAG architectural changes are needed, developing engineering judgment about scope, cost, and security implications; they explore defensive prompting and prompt injection risks.

3. Module 3 — GenAI Architectures: Students discuss the interpretability of transformer models and the reliability of attention as an explanation mechanism, reflecting on what 'understanding' means for a model that generates statistically plausible but potentially factually incorrect text.

4. Module 4 — Language Models & LLMs: Students analyze LLM limitations — training data

biases, knowledge cutoffs, and the critical importance of grounding outputs in verified sources rather than accepting plausible-sounding content uncritically.

5. Module 5 — AI Agents: Students discuss when autonomous agents should defer to human judgment, the alignment problem in agent design, and the risks of uncoordinated multi-agent systems; they propose human-in-the-loop safeguards and define trust boundaries for autonomous deployment.

AI Integration :

1. Module 1 — Introduction to GenAI: Students explore popular generative AI tools (ChatGPT, Claude, DALL-E, Stable Diffusion), categorize them by architecture and modality, and compare outputs across generative model types.

2. Module 2 — Prompt Engineering: Students complete DLI hands-on labs on HuggingFace Transformers, encoder/decoder models, LangChain state management, and RAG fundamentals; classroom session debriefs results and compares strategies across zero-shot, few-shot, and chain-of-thought techniques.

3. Module 3 — GenAI Architectures: Students visualize attention weights and latent spaces using BertViz; compare BERT, GPT-2, and T5 outputs on the same task; run Stable Diffusion with varying CFG scales and ControlNet conditioning to understand diffusion model behavior.

4. Module 4 — Language Models & LLMs: Students compare multiple LLMs (GPT-4, Claude, LLaMA, Gemini) on standardized tasks using perplexity, BLEU, and ROUGE; DLI course on Evaluating RAG and Semantic Search provides quantitative retrieval and deployment evaluation skills.

5. Module 5 — AI Agents: Students build an AI agent using LangChain or CrewAI demonstrating planning, memory-augmented reasoning, and tool use; DLI course on Techniques for Improving RAG provides advanced retrieval patterns for robust agent grounding.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the fundamental concepts, hierarchy, and applications of Generative AI across modalities.	L2
2	Apply prompt engineering strategies (zero-shot, few-shot, chain-of-thought) for effective interaction with generative models.	L4
3	Analyze generative model architectures including transformers, GPT, BERT, T5, and diffusion models.	L4
4	Evaluate different large language models on standardized tasks using perplexity, BLEU, ROUGE, and human evaluation.	L3
5	Design AI agent workflows with planning, memory, and tool use for autonomous task execution using LangChain/CrewAI.	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs														
COS	**Program Outcomes (POs)**											**PSOs**		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO-1	3	2	–	–	2	–	–	2	1	–	–	3	2	2
CO-2	3	3	1	–	3	–	–	–	1	–	–	2	3	2
CO-3	3	3	–	2	2	–	–	–	1	–	–	3	3	2
CO-4	2	2	2	3	2	–	–	2	1	–	–	2	3	3
CO-5	3	2	3	–	3	1	–	–	2	1	2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO-1: Design and build robotic systems for different applications using analytical, logical, and problem-solving skills.
PSO-2: Develop AI-powered software systems to control and operate robotic systems.
PSO-3: Apply robotics and AI to solve real-world problems in a variety of domains, including industrial robotics, service robots, exoskeletons, surgical robots, delivery vehicles, autonomous vehicles, and crewless micro aerial vehicles

TEXT BOOKS:

1. David Foster, *Generative Deep Learning*, 2nd Edition, O'Reilly Media, 2023.
2. Numa Dhamani and Maggie Engler, *Introduction to Generative AI*, Manning Publications, 2024.

REFERENCE BOOKS:

1. Chip Huyen, *AI Engineering: Building Applications with Foundation Models*, O'Reilly Media, 2025.
2. Valentina Alto, *Modern Generative AI with ChatGPT and OpenAI Models*, Packt Publishing, 2023.
3. Ashish Vaswani et al., *Attention Is All You Need*, NeurIPS 2017.

E-RESOURCES:

1. https://huggingface.co/learn/nlp-course (Hugging Face NLP Course)
2. https://www.deeplearning.ai/short-courses/ (DeepLearning.AI LLM & RAG Courses)
3. https://www.promptingguide.ai/ (Prompt Engineering Guide)
4. https://python.langchain.com/ (LangChain Documentation)
5. https://web.stanford.edu/class/cs224n/ (Stanford CS224N: NLP with Deep

Learning)

NVIDIA DLI: Generative AI Explained (2 hrs, FREE) — Foundational overview of GenAI concepts, hierarchy (AI→ML→DL→GenAI), model types (VAE, GAN, diffusion, transformer), domain applications, and ethical challenges. Ideal as pre-reading before Module 1. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-07+V1

NVIDIA DLI: Rapid Application Development With Large Language Models (8 hrs, $90) — Covers HuggingFace Transformers API, encoder and decoder models, LangChain state management, multimodal formulations, and RAG fundamentals with hands-on GPU labs. Reinforces Module 2. URL: https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+C-FX-09+V1

NVIDIA DLI: Introduction to Multimodal Data Curation (1 hr, FREE) — NeMo Curator for processing text, image, and video data at scale; synthetic data pipelines for training and customizing generative models. Supports Module 3. URL: [Verify at learn.nvidia.com]

NVIDIA DLI: Evaluating RAG and Semantic Search Systems (3 hrs, $30) — RAG evaluation metrics (recall, MRR, NDCG, faithfulness), semantic search scoring, enterprise retrieval challenges, and deployment considerations for production systems. Supports Module 4. URL: [Verify at learn.nvidia.com]

NVIDIA DLI: Techniques for Improving the Effectiveness of RAG Systems (4 hrs, $30) — Advanced RAG patterns: query transformation, re-ranking, chunk size optimization, metadata filtering, and evaluation-driven iteration for more reliable agent grounding. Supports Module 5. URL: [Verify at learn.nvidia.com]

RECOMMENDED TOOLS

Python 3.10+, PyTorch, TensorFlow, Hugging Face Transformers
Agent Frameworks: LangChain, LangGraph, CrewAI, AutoGen
LLM APIs: OpenAI, Anthropic Claude, Google Gemini, NVIDIA NIM
Visualization: Matplotlib, BertViz (attention), t-SNE / UMAP
Vector Stores: ChromaDB, FAISS, Pinecone — chromadb.com

LAB EXERCISES

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
MODULE I · Exercises 1–3 : Foundations of Generative AI (06 Hours)					
1	**GenAI Environment Setup & Model Exploration**	Configure Python GenAI stack (Transformers, Diffusers, OpenAI SDK); load pretrained models; explore GPT-2, Stable Diffusion outputs; document differences across architectures.	Python, Transformers, Diffusers, Gradio **Dataset:** *Hugging Face Hub (pretrained checkpoints)*	2 h	**CO1**
2	**Iterative Prompt Engineering Lab**	Design zero-shot, few-shot, and chain-of-thought prompts for the same task; systematically evaluate output quality and hallucination rates; document iteration improvements.	Python, LangChain, OpenAI/Anthropic API **Dataset:** *Diverse task suite (summarization, reasoning, code gen)*	2 h	**CO1**
3	**Simple RAG Pipeline**	Build a simple RAG pipeline with ChromaDB for document retrieval; compare LLM output accuracy before and	LangChain, ChromaDB, Python **Dataset:** *University*	2 h	**CO1**

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
		after grounding with retrieved context.	*FAQ corpus or Wikipedia subset*		
MODULE II · Exercises 4–5 : Transformer Architecture Deep-Dive (04 Hours)					
4	**Attention Mechanism Implementation**	Implement multi-head self-attention from scratch in PyTorch; verify correctness against a reference transformer block.	PyTorch, NumPy **Dataset:** *UCI Adult Income Dataset*	2 h	**CO1**
5	**BERT vs GPT Comparison**	Load pretrained BERT (encoder) and GPT-2 (decoder); compare on classification vs. generation tasks; visualize attention patterns with BertViz.	Hugging Face Transformers, BertViz **Dataset:** *UCI Adult Income Dataset*	2 h	**CO1**
MODULE III · Exercises 6–7 : Diffusion & Multimodal Generation (04 Hours)					
6	**Diffusion Model Inference & Control**	Run Stable Diffusion inference with varying CFG scales and step counts; generate images with ControlNet conditioning; evaluate prompt fidelity.	Diffusers, PyTorch **Dataset:** *Custom prompts + ControlNet reference poses*	2 h	**CO2**
7	**CLIP / BLIP Multimodal Experimentation on Images and Text**	Use CLIP/BLIP for image captioning and visual Q&A; evaluate behavior on ambiguous images and adversarial captions to identify failure modes across modalities.	Transformers, CLIP, BLIP **Dataset:** *COCO Captions subset*	2 h	**CO2**
MODULE IV · Exercises 8–10 : LLM Fine-Tuning & Evaluation (06 Hours)					
8	**LoRA Fine-Tuning of Pretrained LLM**	Fine-tune a 7B-parameter LLM using LoRA on a domain-specific dataset; measure perplexity, task accuracy, and training cost.	PEFT, Transformers, bitsandbytes **Dataset:** *Custom domain corpus (legal / medical / technical)*	2 h	**CO2**
9	**LLM Evaluation Pipeline**	Build an automated pipeline comparing 3 LLMs on a task suite; compute BLEU, ROUGE, perplexity; conduct structured human evaluation.	Hugging Face evaluate, nltk **Dataset:** *Custom task suite (summarization, Q&A, generation)*	2 h	**CO3**
10	**LLM Inference Sizing & Cost Analysis**	Benchmark LLM inference with varying batch sizes, precision (FP16/INT8), and KV-cache strategies; compute cost per 1M tokens.	vLLM, NVIDIA NIM, TensorRT-LLM **Dataset:** *Standard benchmark prompts*	2 h	**CO4**
MODULE V · Exercises 11–12 : AI Agent Design & Capstone (10 Hours)					
11	**AI Agent with Tool Use & Memory**	Build a ReAct agent with LangChain/CrewAI using web search, calculator, and retrieval tools; evaluate reliability; document failure modes.	LangChain, CrewAI, Python **Dataset:** *Multi-hop research questions*	4 h	**CO4**
12	**End-to-End Generative AI Application**	Design a complete GenAI application: problem framing → model selection → prompt engineering → RAG integration →	Full stack: PyTorch, Transformers, LangChain,	6 h	**CO5**

S.No	Exercise Title	Aim / Objective	Tools & Dataset	Hrs	CO
	(Capstone)	agent orchestration → evaluation → NIM deployment.	ChromaDB, NVIDIA NIM **Dataset:** *Student-chosen domain problem*		

CoE (Centre of Excellence) Utilization Mapping for Lab Exercises

DSU AI & Robotics CoE — Dayananda Sagar University | Semester IV | Year 2: Core Systems (2026–2027) | NVIDIA DLI Integration

CoE Note: Lab exercises are aligned with the DSU AI & Robotics CoE Year 2 (Semester IV) Core Systems milestone under the 2026–2029 Blueprint. NVIDIA AI/DLI Lab (DGX) supports 40 students/batch. NVIDIA Jetson Nano = 15–20 units (₹6.36L investment). All NVIDIA DLI courses listed are FREE as per the March 2026 NVIDIA DLI learning path catalogue

DESIGNING MLOPS FOR ENTERPRISES

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3603	**Credits**:	03
Hours / Week	: 3 Hours	**Total Hours**:	39 Hours
L–T–P–S	: 3–0–0–0		

Course Learning Objectives:

This course will enable students to:

1. To impart knowledge on production-level challenges of ML models
2. To provide comprehension of various activities involved in the development, deployment, and monitoring of ML models
3. To familiarize the principles of MLOps and different platforms

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture method*** means it includes not only the traditional lecture method but a different *type of teaching method* that may be adopted to develop the course outcomes.
2. ***Interactive Teaching:*** *Adopt* ***Active learning*** that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show ***Video/animation*** clips to explain the functioning of various concepts.
4. Encourage ***Collaborative*** (Group Learning) Learning in the class.
5. To make ***Critical thinking***, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every ***concept can be applied to the real world*** - and when that's possible, it helps improve the student's understanding.

UNIT – I	**07 Hours**
Introduction to DevOps: SDLC, Virtualization: Containers, Container Orchestration Systems, Cloud platforms, CI/CD: Continuous Integration – Configuration Management, Deployment and Delivery phases, Continuous monitoring, Continuous Testing	
UNIT – II	**08 Hours**
Basic Concepts: Evolution of MLOps, Data-centric AI, ML Development Lifecycle, MLOps Approach, Features of MLOps, ML Data Lifecycle in Production, MLOps maturity levels, ML artifacts, MLOps workflows.	
UNIT – III	**08 Hours**
Machine Learning Pipelines and automation: CI/CD for Machine Learning, ML model serving, Data pipelines, Data drift, ML pipelines: Data ingestion, Feature engineering, Hyperparameter optimization, testing and packaging.	
UNIT – IV	**08 Hours**
Model in MLOps : Model management: Model deployment and monitoring, feedback, orchestration pipelines for ML workflows, ML security, Real-time Streaming ML models, Deployment on edge devices, Automated ML.	
UNIT -V	**08 Hours**
Case Studies on MLOps best practices: **Netflix:** Enhancing Content Recommendations with MLOps, **Uber:** Demand Forecasting with MLOps, **Airbnb**: Search Ranking Models, **Intuit**: Fraud Detection and Prevention, **NASA:** Satellite	

Image Analysis.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain and assess the effectiveness and scalability of an end-to-end machine learning system in a real-world scenario.	L2
2	Recall various metrics used to evaluate machine learning model performance	L2
3	Design and integrate comprehensive automated systems that encompass ML pipelines, CI/CD, data processing, model serving, and drift detection.	L3, L4
4	Analyze the challenges and benefits of implementing MLOps in an organization.	L4
5	Evaluate the applicability and effectiveness of MLOps best practices from case studies in your own or another organization.	L5

Mapping Levels of COs to POs / PSOs														
Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO2	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO3	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO4	2	2	2	2	-	2	-	-	2	1	-	2	2	2
CO5	2	1	-	2	-	-	-	2	-	1	-	-	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

TEXT BOOKS:

1. Alla, Sridhar, and Suman Kalyan Adari. *Beginning MLOps with MLFlow*. Apress, 2021.

2. Rao, Dattaraj. *Keras to Kubernetes: The Journey of a Machine Learning Model to Production*. John Wiley & Sons, 2019.
3. Sculley, David, et al. "Machine learning: The high interest credit card of technical debt." (2014).
4. Jez Humble, David Farley. Continuous Delivery.,2011

REFERENCE BOOKS:

1. Treveil, Mark, Nicolas Omont, Clément Stenac, Kenji Lefevre, Du Phan, Joachim Zentici, Adrien Lavoillotte, Makoto Miyazaki, and Lynn Heidmann. *Introducing MLOps*. O'Reilly Media, 2020.
2. Burkov, Andriy. *Machine Learning Engineering.* True Positive Inc. , 2020.
3. Ameisen, Emmanuel. *Building Machine Learning Powered Applications.* O'Reilly Media, 2020..

Activity Based Learning (Suggested Activities in Class)

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity)
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

****

MINOR PROJECT

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3604	**Credits**	: 02
Hours / Week	: 02	**Total Hours**	: 26 Hours
L–T–P–S	: 0–0–0–4		

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To detail description** of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature.

7. **To analyze** and synthesize the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.

All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.

The following criteria will be checked by the department chairman to approve for the research proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document for the chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team

1. To be dedicated and committed to work on a new research topic by learning new technical skills
2. To have fair knowledge on what is product development or research topic
3. To have constant interaction with allocated guide by providing weekly updates
4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:
There will be CIA evaluation as well as the Semester end evaluation of the work done. Itwill be done by a committee of senior researchers of the Department.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2
3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs

Cos				Program Outcomes (POs)									PSOs	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

LLM based Agent Frameworks
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER-VI

Course Code	: 24AM3606	**Credits**	: 03
Hours/Week	: 03	**Total Hours**	: 45 (15 hrs for Project)
L–T–P–J	: 2–0–0–2	**CIE**	: 100 Marks

Course Vision:
To empower learners with strong foundations and practical expertise in Large Language Models and agent-based AI systems, enabling them to design, analyze, and deploy intelligent, scalable, and efficient AI solutions for real-world applications.

Course Objectives:
This course will enable students to:

1. To understand the architecture and working principles of Large Language Models and transformers
2. To master practical techniques for building AI applications including prompt engineering and RAG
3. To implement agentic workflows and fine-tune models using parameter-efficient techniques
4. To design evaluation methodologies and optimize models for production deployment
5. To develop end-to-end AI engineering skills for real-world LLM applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and role playing.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE–I	**09 Hours**

INTRODUCTION TO AI ENGINEERING AND FOUNDATION MODELS:

Topics: The Rise of AI Engineering: From language models to large language models, from LLMs to foundation models, AI engineering vs ML engineering, the AI engineering stack. **Understanding Foundation Models**: Training data (Speech processing and domain-specific models), model size considerations, post-training (supervised fine-tuning, preference fine-tuning), sampling fundamentals and strategies, test-time compute, structured outputs, the probabilistic/deterministic AI. **Planning AI Applications**: Foundation model use cases (coding, writing, education, conversational bots, information aggregation, workflow automation), use case evaluation, setting expectations, milestone planning, cost and latency considerations, model build vs buy decisions, model econometrics.

MODULE – II	09 Hours
TRANSFORMER ARCHITECTURE AND LLM FUNDAMENTALS **Topics: Working with Text Data**: Understanding word embeddings, tokenizing text (byte pair encoding), converting tokens to token IDs, adding special context tokens, data sampling with sliding window, creating token embeddings, encoding word positions. **Attention Mechanisms**: Exploring LCS (Long Common Sequences), capturing data dependencies with attention, self-attention mechanism (with and without trainable weights), computing attention weights, causal attention and masking future words, dropout in attention, multi-head attention. **Transformer Architecture:** Introduction to transformer architecture, encoder-decoder models, GPT architecture deep-dive. Implementing GPT Components: Layer normalization, feed-forward networks with GELU activations, shortcut connections, connecting attention and linear layers in transformer blocks, coding the GPT model, text generation mechanisms.	
MODULE – III	**09 Hours**
RETRIEVAL-AUGMENTED GENERATION & MODEL ORCHESTRATION: **Topics: Prompt Engineering**: Introduction to prompting, in-context learning (zero-shot and few-shot), system prompts vs user prompts, context length and efficiency. **Best Practices**: Writing clear instructions, providing sufficient context, breaking complex tasks into subtasks, chain-of-thought prompting, iterative prompt development, evaluating prompt engineering tools, organizing and versioning prompts. **Defensive Prompting**: Proprietary prompts and reverse engineering, jail breaking and prompt injection attacks, information extraction risks, defenses against attacks. **RAG Fundamentals**: RAG architecture and workflow, when to use RAG vs fine-tuning, RAG limitations and challenges. Retrieval Systems: Dense retrieval (semantic search with embeddings), sparse retrieval (BM25, TF-IDF), hybrid retrieval strategies, vector databases and indexing (Pinecone, Weaviate, Chroma). **Retrieval Optimization**: Query transformation and expansion, re-ranking strategies, chunk size optimization, metadata filtering, evaluation metrics for retrieval (recall, MRR, NDCG), RAG beyond text (multimodal RAG).	
MODULE – IV	**09 Hours**
AGENT SYSTEMS AND FINE-TUNING TECHNIQUES: **Topics: Agent Systems**: Agent overview and architecture, agentic workflows vs traditional pipelines, when to use agents. Agent Components: Tools and tool use (web search, calculators, databases), planning strategies (ReAct, tree-of-thoughts, chain-of-thought), agent memory (short-term and long-term), agent orchestration frameworks (LangChain, AutoGPT, AutoGEN). **Agent Challenges**: Failure modes and debugging, agent evaluation strategies, cost and latency optimization. **Fine-tuning Fundamentals**: When to fine-tune (vs RAG vs prompt engineering). **Parameter-Efficient Fine-Tuning (PEFT)**: LoRA (Low-Rank Adaptation) theory and implementation, QLoRA for memory efficiency.	
MODULE – V	**09 Hours**
MODEL EVALUATION, INFERING AND DEPLOYMENT: **Topics: Evaluation Methodology**: Challenges of evaluating foundation models, language modeling metrics (entropy, cross-entropy, perplexity, bits-per-character), exact evaluation methods, functional correctness, similarity measurements (lexical and semantic), introduction to embeddings. **AI as a Judge**: Why and how to use AI as judge, limitations, comparative evaluation and ranking models. **Evaluation Criteria**: Domain-specific capability, generation capability, instruction-following capability, cost and latency evaluation. **Designing Evaluation Pipelines**: Model selection workflow, navigating public benchmarks, evaluating all components in a system,	

creating evaluation guidelines, defining evaluation methods and data. **Inference Optimization**: Understanding inference optimization, inference overview, performance metrics (latency, throughput, TTFT, cost), AI accelerators (GPUs, TPUs). **Model Optimization**: Quantization (post-training and quantization-aware), pruning, knowledge distillation. **Inference Service Optimization**: Batching strategies, caching mechanisms, and model serving frameworks.

Course Outcome

CourseOutcomes (COs)	Description	Bloom'sTaxonomy Level
At the end of the course the student will be able to:		
1	Explain the architecture of transformers and GPT models including attention mechanisms.	L2
2	Implement prompt engineering techniques and build RAG systems with retrieval optimization	L3
3	Design and implement agentic workflows and apply parameter-efficient fine-tuning (LoRA, PEFT).	L5
4	Develop comprehensive evaluation methodologies and pipelines for AI systems.	L5
5	Optimize LLM inference and deploy production-ready AI applications with monitoring.	L5

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs

COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	–	1	2	–	–	–	–	–	–	2	1
CO-2	2	3	2	2	3	–	–	–	1	–	–	3	2
CO-3	2	3	3	2	3	–	–	–	2	1	–	3	3
CO-4	2	3	2	3	2	–	–	–	1	1	–	2	3
CO-5	2	3	3	3	3	1	–	–	2	2	1	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration

AI is integrated in this unit through exploratory and application-based learning using tools such as ChatGPT and Google Gemini. Students interact with large language models to understand real-world AI applications, probabilistic outputs, and foundational concepts. Case studies on foundation models are discussed, and learners identify domain-specific AI use cases in areas like education, automation, and conversational systems, enabling conceptual clarity and practical exposure.

This unit incorporates AI integration through hands-on implementation and experimentation with transformer models using libraries such as Hugging Face Transformers. Students perform tokenization, embedding generation, and attention visualization through coding exercises. They also experiment with pre-trained models to understand text generation and internal working of GPT architectures, thereby strengthening their foundational and analytical skills.

This unit integrates AI through the development of intelligent agent systems using frameworks such as LangChain and AutoGPT. Students build agentic workflows capable of tool usage, planning, and memory handling. Additionally, they explore parameter-efficient fine-tuning concepts like LoRA through guided experiments, along with analysing real-world challenges in agent performance and debugging.

AI integration in this unit emphasizes evaluation and deployment of AI systems in practical environments. Students apply evaluation metrics, use AI-as-a-judge techniques, and analyze model performance in terms of latency and cost. They also gain exposure to deploying AI models using APIs and cloud platforms, and perform optimization techniques such as quantization and caching. A mini-project involving end-to-end AI application deployment ensures industry readiness and hands-on expertise.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Chip Huyen, AI Engineering: Building Applications with Foundation Models, O’Reilly ,Media, 2025.

REFERENCE BOOKS:

1. Sebastian Raschka, Build a Large Language Model (From Scratch), Manning Publications, 2025.
2. Jay Alammar and Maarten Grootendorst, Hands-On Large Language Models, O’Reilly Media, 2024.

3. Lewis Tunstall, Leandro von Werra, and Thomas Wolf, Natural Language Processing with Transformers, O'Reilly Media, 2022.
4. Chip Huyen, Designing Machine Learning Systems, O'Reilly Media, 2022.
5. Ashish Vaswani et al., Attention Is All You Need, NeurIPS, 2017.

E-RESOURCES:

Hugging Face NLP Course: https://huggingface.co/learn/nlp-course
DeepLearning.AI Courses on LLMs, RAG, and Agents: https://www.deeplearning.ai/short-courses/
Stanford CS224N: Natural Language Processing with Deep Learning:
https://web.stanford.edu/class/cs224n/

LangChain Documentation: https://python.langchain.com/
OpenAI API Documentation: https://platform.openai.com/docs
Papers with Code – Transformers: https://paperswithcode.com/methods/category/transformers
AI Engineering Resources by Chip Huyen: https://github.com/chiphuyen/aie-book

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, Hugging Face Transformers, LangChain, OpenAI API, Chroma, Pinecone, PyTorch, Weights & Biases.

VISUAL COMPUTING AND GRAPHICS
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER - VI

Course Code	:	24AM3607	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–J	:	2–0–0–2	**CIE**	:	100 Marks

Course Vision:
This course develops an understanding of modern computer graphics by integrating concepts from rendering pipelines, GPU programming, real-time ray tracing, and neural rendering, enabling students to design and analyze advanced visual computing systems.

Course Objectives:

Students will:

1. Understand core concepts of computer graphics and rendering pipelines
2. Analyze geometric transformations and GPU-based rendering techniques
3. Explore lighting, shading, and material models for realistic rendering
4. Evaluate real-time ray tracing and optimization techniques
5. Assess integration of AI/ML in modern graphics systems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method combined with interactive discussions
2. Brainstorming, group activities, and case-based learning
3. Demonstrations using audio-visual tools and datasets
4. Problem-solving sessions and analytical exercises
5. Encouraging critical thinking through real-world scenarios and design challenges

UNIT – I	**08 Hours**

Graphics Pipeline and GPU Fundamentals

Topics:

Introduction to computer graphics, applications in visualization and game engines, graphics rendering pipeline, rasterization vs ray tracing, GPU architecture fundamentals, memory hierarchy, parallel processing, graphics APIs (OpenGL, Vulkan), mathematical foundations (vectors, matrices, coordinate systems).
Applications: Game engines, simulation systems, visualization platforms, real-time rendering systems.

UNIT – II	**09 Hours**

Geometric Transformations and Shader Programming

Topics: 2D and 3D transformations (translation, rotation, scaling), homogeneous coordinates, viewing transformations, projection systems (orthographic and perspective), camera systems, GPU pipeline execution, shader programming (vertex and fragment shaders), texture mapping basics. **Applications:** 3D modeling systems, animation pipelines, AR/VR environments, interactive graphics applications.	
UNIT – III	**06 Hours**
Lighting, Shading and Rendering Techniques **Topics:** Lighting models (ambient, diffuse, specular), Phong and Blinn-Phong shading, physically-based rendering (PBR), normal mapping, shadow mapping, ray tracing fundamentals, BVH acceleration, global illumination basics. **Applications:** Realistic rendering systems, visual effects, gaming graphics, cinematic visualization.	
UNIT – IV	**11 Hours**
Advanced Rendering and Neural Graphics **Topics:** Real-time ray tracing, hybrid rendering, denoising techniques, rendering optimization (culling, LOD), neural rendering basics, NeRF, AI-based denoising, deep learning in graphics, 3D visualization for AI applications. **Applications:** Next-gen game engines, film production, AI-based rendering systems, immersive visualization technologies.	
UNIT – V	**11 Hours**
Advanced Rendering and Neural Graphics **Topics:** Real-time ray tracing, hybrid rendering, denoising techniques, rendering optimization (culling, level-of-detail), neural rendering basics, NeRF, AI-based denoising, deep learning in graphics, 3D visualization for AI applications. **Applications:** Next-generation game engines, film production, AI-based rendering systems, immersive visualization technologies.	

AI Integration:

- AI-assisted visualization and rendering optimization tools
- AI-supported shader generation and transformation modeling
- AI-based lighting and rendering enhancement tools
- Neural rendering models, AI-based denoising, NeRF systems.

Authentic Intelligence:

- Limitations of automated rendering systems
- Errors in shader design and transformation inaccuracies
- Trade-offs between realism and performance
- Ethical use of AI-generated graphics and synthetic content.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain graphics pipeline and GPU architecture concepts	L3
2	Analyze transformations, viewing systems, and shader programming	L4
3	Apply lighting models and shading technique	L3
4	Evaluate real-time ray tracing and optimization methods	L5
5	Evaluate neural rendering and AI integration in graphics	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COS	Program Outcomes (POs)											PSO	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	1	–	–	–	–	–	–	–	–	–	3	2
CO-2	3	3	2	2	2	–	–	–	–	–	–	3	3
CO-3	2	3	2	2	2	–	–	–	–	–	–	3	3
CO-4	2	3	2	2	3	–	–	–	1	–	–	3	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Shirley, Peter, Michael Ashikhmin, and Steve Marschner. *Fundamentals of Computer Graphics*. **4th Edition. A K Peters/CRC Press, 2021.**

2. Akenine-Möller, Tomas, Eric Haines, and Naty Hoffman. *Real-Time Rendering*. **4th Edition. CRC Press, 2018.**
3. Shreiner, Dave and Bill Licea-Kane. *OpenGL Programming Guide: The Official Guide to Learning OpenGL*. **9th Edition. Addison-Wesley, 2013.**

REFERENCE BOOKS:

1. Foley, James D., Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. **3rd Edition. Addison-Wesley, 2013.**
2. Buss, Samuel R. *3D Computer Graphics: A Mathematical Introduction with OpenGL*. **1st Edition, 2003.**
3. Watt, Alan. *3D Computer Graphics*. **3rd Edition. Addison-Wesley, 2000.E-**

RESOURCES:

- Khronos Group OpenGL Documentation: https://www.khronos.org/opengl/
- Vulkan API Specifications: https://www.khronos.org/vulkan/
- NVIDIA Developer Blog – Ray Tracing, DLSS, and Neural Rendering: https://developer.nvidia.com/blog/
- SIGGRAPH Papers and Course Materials: https://kesen.realtimerendering.com/
- High-Performance Graphics Conference: https://www.highperformancegraphics.org/
- Global Illumination Compendium: https://people.cs.kuleuven.be/~philip.dutre/GI/
- NPTEL Computer Graphics: https://nptel.ac.in/courses/106106090/

RECOMMENDED TOOLS

- **Programming:** C++ / Python
- **Graphics APIs:** OpenGL / Vulkan
- **Frameworks:** Unity / Unreal Engine (conceptual exposure)
- **Libraries:** OpenGL Shading Language (GLSL)
- **AI Tools:** ChatGPT, AI-based rendering tools

ROBOT OPERATING SYSTEMS (ROS)

Professional Elective – II
SEMESTER - VI

Subject Code	: 24AM3608	**Credits**	:	03
Hours / Week	: 03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S	: 2–0–0–2	**CIE :** 100 Marks		

Course Vision:
To develop the ability to design, implement, and integrate robotic systems using ROS middleware by combining communication, perception, and control components. The course aims to provide a strong system-level understanding of robotic software architecture, enabling students to build scalable and modular robotic applications aligned with modern autonomous and intelligent systems.

Course Learning Objectives:

Students will:

1. Understand ROS architecture and communication mechanisms for robotic systems.
2. Develop ROS-based programs using nodes, topics, and services.
3. Implement sensor integration and data handling in robotic applications.
4. Analyze and debug robotic systems using ROS tools and visualization frameworks.
5. Build and simulate robotic systems using ROS and simulation environments.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe ROS architecture and communication mechanisms.	L4
CO2	Develop ROS-based programs using nodes, topics, and services.	L5
CO3	Evaluate sensor integration and data processing in ROS-based systems.	L5
CO4	Analyze and debug robotic systems using ROS visualization and diagnostic tools.	L4
CO5	Design and implement a basic robotic system using ROS in simulation or hardware.	L5

UNIT – I : Introduction to ROS and Robotic Middleware	**07 Hours**

Evolution of robotic middleware, ROS ecosystem, computation graph, nodes, topics, services, ROS architecture, ROS file system, basic ROS setup.
Applications: Autonomous robots, service robots

UNIT – II : ROS Programming and Communication	**07 Hours**
ROS workspace and package creation, publishing and subscribing, message types, services, ROS client libraries (Python), inter-node communication. Applications: Sensor data communication, robot control pipelines	
UNIT – III : Sensor Integration and Data Handling	**08 Hours**
Integration of sensors (camera, LiDAR – conceptual), data acquisition in ROS, message handling, basic data processing, introduction to sensor-based decision pipelines. Applications: Perception-enabled robotic systems	
UNIT – IV : ROS Tools, Simulation, and System Integration	**08 Hours**
ROS debugging tools (rosnode, rqt_graph, rosout, rqt_console), RViz visualization, rosbag data logging, introduction to Gazebo simulation, basic robot simulation, teleoperation, introduction to navigation concepts in ROS. Applications: Autonomous navigation systems, warehouse robots	
UNIT – V :NVIDIA DLI: ROS 2 and Isaac Sim	**15 Hours**
Getting Started with Issac Sim - Building Your First Robot in Isaac Sim, Ingesting Robot Assets and Simulating Your Robot in Isaac Sim, Synthetic Data Generation for Perception Model Training in Isaac Sim, Developing Robots With Software-in-the-Loop (SIL) in Isaac Sim, Leveraging ROS 2 and Hardware-in-the-Loop in Isaac Si Getting Started With Isaac ROS - An Introduction to AI-Based Robot Development With Isaac ROS Accelerating ROS 2 Workloads With NVIDIA GPU-Powered Libraries and AI Models -	

Mapping Levels of COs to POs / PSOs													
COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2

	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and World	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	-	2	-	-	-	-	-	1	3	1
CO2	3	3	2	1	3	-	-	1	-	-	2	3	3
CO3	3	3	2	2	3	-	-	1	-	-	2	3	3
CO4	3	3	2	2	3	-	1	2	1	1	2	3	3
CO5	3	3	3	2	3	-	1	2	2	1	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low) AI**

Integration & Authentic Intelligence Reflection (AI² Pedagogy)

AI Integration:

ROS as middleware for AI-enabled robotic systems

Integration of perception modules into ROS

Handling perception data in ROS pipelines

Simulation-based development for AI-driven robotics

Authentic Intelligence Reflection:

Role of human supervision in robotic autonomy

Reliability and robustness in communication systems

Handling noisy and uncertain sensor data

Simulation-to-real challenges and validation

Mini Project (Mandatory)

Students will design and implement a ROS-based robotic system (simulation or hardware-based).

Minimum Requirement (Mandatory for all)

- ROS-based system with:
  - Sensor input
  - Inter-node communication
  - Basic control or decision logic

Example Problem Statements:

- ROS-based obstacle avoidance robot

- Sensor-driven mobile robot simulation
- ROS-based teleoperation with feedback
- Basic perception-integrated robotic system

Advanced Track (Optional / High-grade band)

- Integration with SLAM or navigation stack
- Multi-node system with modular design

Evaluation Focus:

- System design and architecture
- Correct ROS implementation
- Integration quality
- Debugging and validation

Recommended Tools

- ROS2
- Gazebo
- RViz
- Python
- Ubuntu Linux environment

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding

- Students may be asked to explain or modify AI-assisted solutions

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

Text Books

1. Quigley, Morgan; Gerkey, Brian; Smart, William D., Programming Robots with ROS, 1st Edition, O'Reilly Media, 2015.
2. Joseph, Lentin, Robot Operating System for Absolute Beginners, 1st Edition, Apress, 2018.

Reference Books

1. Fairchild, Carol & Harman, Thomas L., ROS Robotics by Example, 2nd Edition, Packt Publishing, 2017.
2. Martinez & Fernandez, Mastering ROS for Robotics Programming, 3rd Edition, Packt, 2021.
3. Bassa, Marco Matteo, ROS2 Best Practices, 1st Edition, 2023.

E-Resources

1. https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-sim/latest/index.html
2. https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-ros/latest/index.html
3. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-OV-02+V1

****

AI FOR AUTONOMOUS VEHICLES & SMART MOBILITY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	:	24AM3609	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S		2–0–0–2	**CIE**	:	100 Marks

Course Learning Objectives:

This course will enable students to:

1. Understand AI-driven architecture of autonomous vehicles and smart mobility systems.
2. Apply machine learning and deep learning techniques for perception, prediction, and decision-making.
3. Analyze localization, planning, and control using AI-based approaches.
4. Develop AI-enabled solutions for real-world smart mobility applications.
5. Evaluate safety, ethics, and societal impacts of AI in transportation systems.

Course Vision

AI for Autonomous Vehicles & Smart Mobility introduces students to the integration of artificial intelligence with modern transportation systems. The course covers perception using deep learning, decision-making using machine learning models, and intelligent path planning for autonomous navigation.

Course Outcome

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain AI-based architecture of autonomous vehicles	L2
2	Apply ML/DL models for perception and prediction tasks	L3
3	Analyze AI-driven localization and path planning techniques	L4
4	Develop intelligent mobility solutions using AI	L3
5	Analyze ethical, safety, and societal implications of AI systems	L4

Course Modules :

Module I : AI in Smart Mobility & Autonomous Systems	**09 Hours**

Evolution of mobility, levels of automation, AV architecture, Role of AI & ML in daily transportation, introduction to AI in transportation, intelligent transportation systems, AI pipeline (data → model → decision).

Applications: Self-driving cars, smart cities, ride-sharing platforms.

Module II : AI-Based Perception Systems	**09 Hours**

Sensors (Camera, LiDAR, RADAR), sensor fusion, Computer Vision basics for autonomous vehicles, image processing, Convolutional Neural Networks (CNNs), object detection (YOLO, SSD), semantic segmentation. ***Applications:*** Lane detection, obstacle detection, and pedestrian recognition under different environmental scenario.

UNIT – III : AI for Localization & Prediction	**09 Hours**

Simultaneous Localization and Mapping (SLAM), neural networks, supervised learning: regression & classification in mobility systems, time-series prediction, traffic forecasting. ***Applications:*** GPS enhancement, traffic prediction, recommendations, and route optimization.
UNIT – IV : AI-Based Planning, Control & Connectivity **09 Hours**
Path planning (Dijkstra, A*), reinforcement learning, decision-making models, trajectory planning, V2X communication, Edge AI in smart mobility, 5G mobility**,** Real-life examples: Smart traffic signals, emergency vehicle prioritization. ***Applications***: Autonomous navigation, smart traffic signals, connected vehicles, V2G, G2V.
UNIT – V : AI Safety, Ethics & Future Mobility **09 Hours**
Safety challenges in autonomous vehicles, Explainable AI (XAI), safety in AI systems, bias in ML models, ethical decision-making, legal frameworks, sustainability. Future trends: Electric vehicles, shared mobility, AI-based smart city ecosystems. ***Applications:*** Autonomous cars, delivery robots, smart city ecosystems, and personalized recommendations.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and the world	Ethics	Individual & Collaborative Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	-	2	-	1	-	-	-	2	1	1
CO2	3	3	2	1	3	-	-	-	-	-	2	2	2
CO3	3	3	3	2	3	-	-	-	-	-	2	2	2
CO4	3	3	3	2	3	1	1	2	1	-	2	2	3
CO5	3	3	2	2	2	2	3	1	1	1	3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- AI Integration:
  - Module-1: AI-driven mobility system analysis
  - Module-2: Deep learning for perception (CNN models)
  - Module-3: ML models for prediction and forecasting
  - Module-4: Reinforcement learning for navigation

- Module-5: Explainable AI and ethical AI systems.

- Authentic Intelligence Reflection:
  - Module-1: Evaluate reliability of AI perception models
  - Module-2: Identify bias in datasets
  - Module-3: Analyze safety-critical decisions
  - Module-4: Reflect on ethical dilemmas
  - Module-5: Assess sustainability in AI mobility.

RESEARCH INTEGRATION TEACHING

1. Paper review on autonomous driving datasets (KITTI, Waymo, TiAND).
2. Case studies on smart mobility systems.
3. Mini presentations on emerging AI mobility technologies.
4. AI-assisted research using tools for analysis.
5. Critical evaluation of safety and ethical issues.

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
   a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
   b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
   c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI² Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI Perception Model Evaluation

Students prompt an AI tool:

"" Generate code for lane detection using CNN." Students must:

1. Analyze accuracy
2. dataset limitations
3. improve performance

Assignment 2: Traffic Prediction Model Verification

- Students ask AI:

  " Compare ML models (Linear Regression vs Neural Networks)."

- Students must verify:

  1. Evaluate accuracy and edge cases

Assignment 3: AI Debugging Challenge

- Debug faulty object detection or prediction model
- Students must:

  1. Identify the bug
  2. Fix the code
  3. Explain the error

Assignment 4: Mobility Data Visualization

- Students use AI visualization tools to:

1. Analyze real-world traffic datasets

Assignment 5: AI-Based Path Planning Study

1. Compare classical algorithms vs AI-based approaches

Suggested Reading:

Primary Textbooks

1. Mahajan, Rupali A., Rajesh Dey, Parikshit N. Mahalle, and Vivek S. Deshpande, AI-Driven Transportation Systems: Real-Time Applications & Related Technologies. Springer, 2025.
2. Houbing Song, Sabina Jeschke, et al., Smart Transportation: AI Enabled Mobility and Autonomous Driving. CRC Press, 2021.

3. Dayyala, N. M. Kottayi, and R. B. Mallick, Machine Learning in Transportation: Applications with Examples and Codes. De Gruyter, 2024.

Additional Resources

1. IEEE research articles
2. Hussein T. Mouftah, Melike Erol-Kantarci, Sameh Sorour, onnected and Autonomous Vehicles in Smart Cities. CRC Press, 2020.
3. Dominique Paret and Hassina Rebaine, Autonomous and Connected Vehicles: Network Architectures from Legacy Networks to Automotive Ethernet. Wiley, 2022.
4. Y. Wang, Z. Cui, and R. Ke, Machine Learning for Transportation Research and Applications. Elsevier, 2023.

E-Resources : (Recommended Tools)

- Python/OpenCV
- Tensorflow/ PyTorch
- Scikit-learn
- CARLA Simulator, SUMO Traffic Simulator
- Jupyter Notebook

Activity Based Learning (Suggested Activities in Class)

1. Group discussion: Discuss different AI techniques used in autonomous vehicles such as perception, localization, and path planning to understand real-world mobility challenges.
2. Collaborative Activity: Develop a mini project in teams of 4 students, such as a smart traffic system, lane detection model, or traffic prediction system using AI techniques.

****

CONVERSATIONAL AI AND SPOKEN LANGUAGE UNDERSTANDING

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3610	**Credits**:	03
Hours / Week	: 04 Hours	**Total Hours**:	45 (15 hrs for Project)
L–T–P–S	: 2–0-0–2	**CIE :** 100 Marks	

Prerequisites

Fundamentals of Artificial Intelligence, Machine Learning concepts, Probability and Statistics, Natural Language Processing. Familiarity with Python programming language is desirable. Prior exposure to optimization or deep learning and Neural Network concepts will be an added advantage.

Course Learning Objectives:

This course will enable students to:

1. Understand classical and modern Conversational AI architectures, including LLM-based systems.
2. Apply NLP and deep learning techniques for language understanding and response generation.
3. Design intelligent dialogue systems using retrieval, generation, and reinforcement learning.
4. Develop, evaluate, and deploy scalable conversational AI applications.
5. Analyse ethical, societal, and research challenges in advanced conversational systems.

UNIT – I : Foundations of Conversational AI	**8 Hours**
Evolution of conversational systems, Types of conversational systems: task-oriented, open-domain, hybrid, Core components of the system, Understanding Conversational AI vs Chatbots vs Virtual Assistants, Applications of conversational systems, Challenges in conversational systems.	
UNIT – II : Natural Language Understanding & Representation Learning	**8 Hours**
Text preprocessing and linguistic features, Intent classification and entity recognition, Machine learning approaches for NLU, Word embeddings: Word2Vec, GloVe, FastText, Contextual embeddings: BERT, RoBERTa, Transformer architecture for NLU, Context handling, Ambiguity resolution	
UNIT – III : Dialogue Management, Generation & LLMs Dialogue systems: task-oriented vs open-domain, Dialogue state tracking, Rule-based dialogue management, Machine learning-based dialogue management, Natural Language Generation (NLG): template-	**8 Hours**

based and model-based approaches, Introduction to Large Language Models, Prompt engineering and in-context learning, Retrieval-Augmented Generation (RAG) for grounded conversations, Handling errors, fallbacks and mitigation strategies.	
UNIT – IV : Advanced Conversational AI Systems	**8 Hours**
Knowledge-grounded dialogue systems, Conversational Information Retrieval, Emotion-aware and empathetic conversational agents, Multimodal Conversational AI (text, speech, vision), Reinforcement Learning and RLHF for dialogue optimization, Explainable Conversational AI (XAI)	
UNIT – V : Deployment, Evaluation, Ethics & Future Trends	**7 Hours**
Evaluation metrics, human evaluation of conversational agents, deployment architectures and APIs, MLOps for conversational AI, security issues, future research directions and open challenges	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Explain classical and modern conversational AI architectures	L2
2	Apply NLP and transformer-based models for language understanding	L3
3	Design dialogue systems using retrieval, generation, and RL	L4
4	Implement and evaluate conversational AI applications	L3
5	Analyze ethical, societal, and research challenges in conversational AI	L5

Table: Mapping Levels of COs to POs / PSOs														
COs	Program Outcomes (POs)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CO1	3	2	1	2	1							2	2	1
CO2	2	3	2	3	2							2	3	2
CO3	2	3	3	3	2				1	1	_	2	3	3
CO4	1	2	3	3	3				2	2	1	3	3	3
CO5	1	2	2	3	2	1	1	3	2	2	2	3	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. **Romil Rawat, Rajesh Kumar Chakrawarti, Sanjaya Kumar Sarangi, Piyush Vyas, Mary Sowjanya Alamanda, Kotagiri Srividya, Krishnan Sakthidasan Sankaran "Conversational Artificial Intelligence", Scrivener Publishing, January 2024.**
2. **Michael McTear , Marina Ashurkina , "Transforming Conversational AI", February 2024.**

REFERENCE BOOKS:

1. **"Effective Conversational AI" by Andrew Freed, Cari Jacobs, Eniko Rozsa, Foreword by Jesús Mantas, April 2025.**
2. "Designing Bots: Creating Conversational Experiences" by Amir Shewat, June 2017.
3. **The Definitive Guide to Conversational AI with Dialogflow and Google Cloud by Lee Boonstra, 2021.**

E-Resources:

1. **Conversational AI: LLM Driven Development for Beginners – course by udemy**
2. **Getting started with Conversational AI - https://www.youtube.com/watch?v=OsbY7juoKgI**
3. **Conversational AI (2024): A Comprehensive Guide - https://www.gupshup.ai/resources/guide/conversational-ai-comprehensive-guide/**

MULTI AGENT AI SYSTEM

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VI

Course Code	: 24AM3611	**Credits**	: 03
Hours/Week	: 03	**Total Hours**	: 45 (15 hrs for Project)
L–T–P–J	: 2–0–0–2	**CIE+SEE**	: 100 Marks

Course Vision:

To develop learners' understanding of intelligent and multi-agent systems, enabling them to analyze, design, and apply autonomous agent-based solutions for complex real-world problems with consideration of coordination, communication, and ethical implications.

Course Objectives:

This course will enable students to:

1. Understanding the principles behind intelligent agents and agent-oriented systems.
2. Knowledge of architectural models enabling autonomous agent behaviour.
3. Familiarity with cooperation, coordination, and communication in multi-agent societies.
4. Awareness of ethical implications and governance challenges of autonomous agents.
5. Capability to conceptualize and build agent-based solutions for practical problems.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, note taking, annotating, and role playing.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE–I	**09 Hours**

Intelligent Agents vs Conventional Expert Systems

Topics: Agents and objects, agents versus expert systems, agents versus distributed systems, characteristics of intelligent agents, agent environments and environment properties, typical application domains of agent systems.

MODULE – II	**09 Hours**

Intelligent Agent Architectures

Topics: Abstract architectures for intelligent agents, task environments and agent functions, design of intelligent agents, reasoning agents and BDI architecture, reactive agents and subsumption architecture, hybrid agents such as PRS architecture, Multi-layered agent architectures including InterRAP.

MODULE – III	09 Hours
Multi-Agent Systems and Interaction Models **Topics:** Definition and characteristics of multi-agent systems, classification of multi-agent interactions, cooperative versus non-cooperative agents, zero-sum and non-zero-sum interactions, Prisoner's Dilemma, Axelrod's experiments on cooperation, self-interested agents and strategic interaction.	
MODULE – IV	**09 Hours**
Agent AI, Communication, Coordination and Ethics **Topics:** Principles of agent communication, speech acts and communication semantics, agent communication languages, Knowledge Query and Manipulation Language (KQML), Knowledge Interchange Format (KIF), FIPA framework and interaction protocols, ontologies for agent communication, ethical issues in autonomous agent communication, accountability and transparency in agent interactions .	
MODULE – V	**09 Hours**
Cooperative Multi-Agent Systems and Ethical Framework **Topics:** Cooperative Distributed Problem Solving (CDPS), Partial Global Planning, coordination and coherence in agent societies, negotiation and argumentation among agents, auction-based multi-agent systems, ethical risks in multi-agent decision-making, governance and societal impact of agent-based systems, real-world applications of intelligent agents and multi-agent systems .	

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain the concepts of intelligent agents and applications suitable for agent-oriented solutions.	L2
2	Describe agent architectures and mechanisms enabling autonomous intelligent behaviour.	L3
3	Analyze interaction and coordination strategies in multi-agent systems.	L4
4	Evaluate ethical, social, and governance challenges in agent-based systems.	L4
5	Design and develop a basic intelligent or multi-agent system for a real-world problem	L3

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	2	–	1	1	–	–	–	–	–	–	2	1
CO-2	2	3	2	2	2	–	–	–	1	–	–	2	2
CO-3	2	3	3	2	2	–	–	–	2	1	–	3	2
CO-4	2	2	1	2	1	2	1	2	1	1	–	2	2
CO-5	2	3	3	2	2	2	1	2	2	2	1	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration:

AI integration in this unit is achieved through conceptual exploration and interactive demonstrations using AI-based tools and simulators. Students analyze real-world intelligent agent applications such as chatbots, recommendation systems, and autonomous assistants. They compare traditional expert systems with modern AI agents through case studies and hands-on interactions, enabling them to understand agent characteristics, environments, and practical application domains..

This unit integrates AI through modeling and simulation of various agent architectures such as BDI, reactive, and hybrid systems. Students engage in practical exercises to design simple agents and observe behavior differences across architectures. Tools and frameworks for agent development are introduced to demonstrate subsumption and layered architectures, helping learners visualize decision-making and autonomy in intelligent systems.

AI integration focuses on simulating multi-agent environments where students observe cooperation, competition, and strategic interactions. Practical activities include modeling scenarios like Prisoner's Dilemma and analyzing agent strategies. Simulation tools are used to study coordination and interaction patterns, enabling students to understand real-world applications such as distributed systems, traffic management, and economic systems.

In this unit, AI integration is achieved through implementation and analysis of agent communication protocols using standard frameworks. Students explore communication languages and simulate agent interactions using message-passing systems. Ethical considerations such as transparency, accountability, and bias in autonomous agents are discussed through case studies, encouraging critical thinking about responsible AI deployment.

AI integration in this unit emphasizes real-world application development and problem-solving using cooperative multi-agent systems. Students design small-scale agent-based solutions for distributed problem solving, negotiation, and coordination scenarios. Practical exposure includes analyzing real-world systems such as smart grids, autonomous vehicles, and e-commerce agents, along with evaluating ethical risks and governance challenges in multi-agent decision-making.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Michael Wooldridge, *An Introduction to MultiAgent Systems*, 2nd Edition, Wiley, 2009.
2. Stuart J. Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 4th Edition, Pearson, 2021.

REFERENCE BOOKS:

1. François Chollet, *Deep Learning with Python*, 2nd Edition, Manning Publications, 2021.
2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 3rd Edition, O'Reilly Media, 2022.

E-RESOURCES:

2. FIPA Agent Standards : https://www.fipa.org
3. MIT OpenCourseWare – Multi-Agent Systems : https://ocw.mit.edu
4. NVIDIA AI Resources: https://developer.nvidia.com/ai
5. Coursera – Multi-Agent Systems & AI Courses: https://www.coursera.org
6. Stanford Online AI Courses: https://online.stanford.edu

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, JADE, SPADE, TensorFlow, PyTorch, NetLogo.

Intelligent Computational Imaging and Reconstruction

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER - VI

Course Code	:	24AM3612	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–J	:	2–0–0–2	**CIE**	:	100 Marks

Course Vision:

To develop the ability to model, analyze, and design advanced imaging systems by integrating physical models, computational methods, and AI-driven techniques in **Computational Imaging**, enabling efficient and intelligent visual data reconstruction.

Course Objectives:

This course will enable students to:

1. Understand image formation and computational imaging principles.
2. Apply inverse problem solving and optimization techniques.
3. Analyze advanced imaging methods such as HDR and super-resolution.
4. Explore modern imaging modalities and sensing systems.
5. Design AI-based imaging solutions for real-world applications.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture method** along with traditional lecture method, different type of teaching methods may be adopted to develop the course outcomes.
2. **Interactive Teaching:** incorporating brainstorming, discussing, group work, focused listening, formulating questions, notetaking, annotating, and roleplaying.
3. Showing **Video/animation** films to explain functioning of various concepts.
4. Encourage **Collaborative** (Group Learning) Learning in the class.
5. To make **Critical thinking**, asking Higher order Thinking questions in the class in the form of Quiz and writing programs with complex solutions.
6. Showing the **different ways to solve** the same problem and encourage the students to come up with their own creative ways to solve them

MODULE – I	**09 Hours**

Foundations of Computational Imaging

Introduction to **Computational Imaging**, imaging system overview, image formation models, radiometry and photometry, sampling and quantization, linear shift-invariant systems, convolution and correlation operations, frequency domain analysis using the **Fourier Transform**. ***Applications:*** Image acquisition and enhancement in medical imaging, computational photography, remote sensing systems, and scientific imaging applications.

MODULE – II	**09 Hours**

Inverse Problems and Optimization in Imaging

Forward and inverse imaging problems, ill-posed problems and their challenges, regularization techniques such as **Tikhonov Regularization**, L1 regularization, and total variation, Bayesian estimation methods, **Maximum A Posteriori (MAP) Estimation** and **Minimum Mean Square Error (MMSE) Estimation**, optimization techniques including **Gradient Descent**, proximal methods, and **Alternating Direction Method of Multipliers (ADMM)**

Applications: Image reconstruction in medical imaging (CT/MRI), deblurring and denoising in photography, super-resolution imaging, and restoration in scientific and remote sensing applications.

MODULE – III	**09 Hours**

Advanced Computational Imaging Techniques

Advanced techniques in **Computational Imaging** including computational photography, high dynamic range imaging using **HDR Imaging**, super-resolution imaging, image deblurring and denoising methods, compressive sensing and sparse representations, coded aperture imaging, and phase retrieval techniques.

Applications: Enhancement of photographic quality in mobile cameras, medical image reconstruction, astronomical imaging, surveillance systems, and high-resolution imaging in scientific research.

MODULE – IV	**09 Hours**

Advanced Imaging Systems and Modalities

Advanced imaging techniques in **Computational Imaging** including light field imaging, lensless and coded imaging systems, computational microscopy, basics of computational tomography such as **Computed Tomography (CT)** and **Magnetic Resonance Imaging (MRI)**, hyperspectral imaging, time-of-flight and depth imaging, and event-based cameras.

Applications: 3D scene reconstruction, biomedical imaging and diagnostics, remote sensing, autonomous systems, and advanced scientific imaging applications.

.

MODULE – V	**09 Hours**

Learning-Based Computational Imaging

Topics: Learning-based approaches in **Computational Imaging**, deep learning techniques for imaging, convolutional neural networks (**Convolutional Neural Networks (CNNs)**) for image reconstruction, plug-and-play priors, physics-informed neural networks, end-to-end imaging system design, self-supervised and unsupervised learning methods for imaging, and ethical and societal considerations in intelligent imaging systems.

Applications: AI-based image reconstruction in medical imaging, intelligent denoising and super-resolution, automated visual inspection systems, and smart imaging solutions in scientific and industrial domains.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		

1	Understand image formation and computational imaging principles.	L2
2	Apply inverse problem solving and optimization techniques	L4
3	Analyze advanced imaging methods such as HDR and super-resolution	L4
4	Explore modern imaging modalities and sensing systems.	L3
5	Design AI-based imaging solutions for real-world applications.	L5

- **AI Integration:**
  - Module-1: Integration of computational imaging with AI techniques for image reconstruction, denoising, and enhancement using machine learning-based models.
  - Module-2: Use of AI and deep learning models for solving inverse imaging problems, data-driven regularization techniques, and intelligent optimization for faster and more accurate image reconstruction.
  - Module-3: Use of deep learning models for super-resolution, AI-based denoising and deblurring, and intelligent reconstruction techniques for compressive sensing and advanced imaging systems.
  - Module-4: Integration of advanced imaging systems with AI for depth estimation, hyperspectral data analysis, medical image interpretation, and real-time intelligent vision systems.
  - Module-5: Advanced use of deep learning models for end-to-end imaging pipelines, integration of physics-based models with AI, and intelligent data-driven reconstruction techniques.

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	2	2	–	2	2	–	3	–	–	2	3	3	3
CO-2	3	3	2	–	3	3	–	3	–	–	–	3	3	3
CO-3	3	2	2	–	3	3	–	3	–	–	–	3	3	3
CO-4	3	2	3	2	3	3	–	3	2	–	–	3	3	3
CO-5	3	–	2	–	3	3	–	3	2	2	2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Veeraraghavan – Computational Imaging
2. Szeliski – Computer Vision

REFERENCE BOOKS:

E-RESOURCES:

1. MIT OpenCourseWare
2. Stanford Algorithms Lectures
3. IEEE research articles

RECOMMENDED TOOLS

Python 3.10+, NumPy, Pandas, Scikit-learn, PyTorch, Hugging Face Transformers
IBM AI Fairness 360 (AIF360) — https://aif360.mybluemix.net
Aequitas Bias Audit Toolkit — http://aequitas.dssg.io
SHAP — https://shap.readthedocs.io | LIME — https://github.com/marcotcr/lime
OpenAI Gymnasium (RL) — https://gymnasium.farama.org

AUTONOMOUS ROBOTIC SYSTEMS
SEMESTER-VI
Professional Elective

Subject Code	:	24AM3613	**Credits**	:	03

Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S	:	2–0–0–2	**CIE :** 100 Marks		

Course Learning Objectives:

Students will:

1. Understand the architecture and design principles of autonomous robotic systems.
2. Apply perception and localization techniques under real-world sensing uncertainty.
3. Analyze trade-offs between accuracy, computation, and robustness in autonomy pipelines.
4. Design planning and navigation strategies under dynamic and constrained environments.
5. Integrate multiple subsystems into a cohesive autonomous system.
6. Evaluate system performance under uncertainty, noise, and failure conditions.

Course Description :

To enable students to design, implement, and evaluate autonomous robotic systems under uncertainty by integrating perception, localization, planning, and system-level decision-making frameworks for real-world deployment.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the architecture of autonomous robotic systems	L4
CO2	Apply perception techniques for environment understanding under uncertainty	L3
CO3	Analyze and implement localization and mapping methods	L4
CO4	Design planning and decision-making strategies for autonomous navigation	L6
CO5	Develop and evaluate an integrated autonomous robotic system	L6

UNIT – I : Perception for Autonomous Systems	**03 Hours**

Sensors for robotics (LiDAR, camera, IMU), data acquisition, feature extraction, object detection (classical + deep learning overview), basic sensor fusion concepts.
Applications: Mobile robots, autonomous vehicles, service robotics

NVIDIA DLI Self-Paced Courses Mapped to Module-1	**06 Hours**

Getting Started with Deep Learning - Module: Image classification using convolutional neural networks

Building Real-Time Video AI Applications - Modules: Object detection in images and video streams, Deep learning inference workflows for visual data

Extension:
- Application of deep learning-based perception in robotic systems
- Integration of object detection outputs into robotic perception pipelines
- Role of real-time perception in autonomous systems

UNIT – II : Localization and Mapping	**09 Hours**

Coordinate frames, odometry, probabilistic localization (Kalman Filter, Particle Filter concepts), mapping basics, occupancy grid maps, SLAM overview (visual and LiDAR-based).
Nvidia DLI: Getting Started with Deep Learning - Modules: Feature extraction using convolutional neural networks, Learned visual representations for image-based tasks
Applications: Indoor navigation, warehouse robotics, mapping unknown environment

UNIT – III : Planning and Decision Making	**09 Hours**

Graph-based planning methods (A*, Dijkstra), sampling-based planning (RRT, PRM), and hierarchical decision-making frameworks incorporating cost functions and constraints.
Applications: Robot navigation, path optimization, exploration tasks

UNIT – IV : Navigation and Execution	**09 Hours**

Navigation stack architecture (ROS-based), global vs local planning, trajectory generation, execution constraints, obstacle avoidance in dynamic environments, system limitations.
Nvidia DLI: Building Real-Time Video AI Applications - Modules: Real-time video inference workflows, Processing visual data streams for object detection

Applications: Autonomous mobile robots, drone navigation

UNIT – V : System Integration and Evaluation	**09 Hours**

System-level architecture, integration of perception–localization–planning–execution, ROS pipelines, simulation-to-real transfer, evaluation metrics and benchmarking, failure case analysis, case studies (autonomous vehicles, warehouse robots).
Nvidia DLI: Building Real-Time Video AI Applications - End-to-end video AI pipelines
- Model inference and deployment workflows

Applications: End-to-end autonomous systems

Mapping Levels of COs to POs / PSOs													
COs				Program Outcomes (POs)								PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and World	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	1	-	1	-	-	-	-	-	1	3	1
CO2	3	3	2	2	3	-	-	1	-	-	2	3	3
CO3	3	3	2	3	2	-	-	1	-	-	2	3	3
CO4	3	3	3	2	3	-	1	2	1	1	2	3	3
CO5	3	3	3	2	3	1	1	2	2	1	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Vision-based perception using deep learning models

Module-2: Learning-based localization (overview only)

Module-3: Reinforcement learning for decision-making (conceptual exposure)

Module-4: Learning-based control (high-level overview)

Module-5: Modular AI pipelines in robotics

Authentic Intelligence Reflection:

Module-1: Limitations of perception systems, uncertainty and noise handling

Module-2: Uncertainty modeling, failure modes in localization

Module-3: Trade-offs between optimality and computational efficiency

Module-4: Safety, robustness, and real-world deployment constraints

Module-5: System reliability, ethical deployment, human-in-the-loop systems.

Mini Project (Mandatory)

Students will design and implement a basic autonomous robotic system (simulation or hardware-based).

Minimum Requirement (Mandatory for all)

- Autonomous navigation with:
- Perception input
- Basic planning
- Obstacle avoidance

Advanced Track (Optional / High-grade band)

- SLAM-based navigation OR
- Dynamic environment decision-making OR
- Vision-based autonomy

Evaluation Focus

- System design clarity
- Level of autonomy achieved
- Robustness under uncertainty
- Failure handling and recovery

Recommended Tools

- ROS / ROS2
- Gazebo / Webots
- Python / C++
- OpenCV
- RViz
- SLAM packages (e.g., Gmapping, Cartographer)

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

Suggested Reading

1. Thrun, S., Burgard, W., Fox, D., Probabilistic Robotics, 2005, MIT Press.
2. LaValle, S. M., Planning Algorithms, 2006, Cambridge University Press.
3. Siciliano, B., Khatib, O., Springer Handbook of Robotics, 2016, Springer.
4. Corke, P., Robotics, Vision and Control, 2017, Springer.
5. Selected recent IEEE/ICRA/IROS research papers on autonomous systems
6. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-01+V1
7. https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-IV-01+V1

AI FOR HEALTHCARE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Course Code	:	24AM3614	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45(15 hrs for Project)
L–T–P-J	:	2-0-0-2	**CIE :**		**100 Marks**

Course Vision: It aims to equip students with skills to apply artificial intelligence in improving diagnosis, treatment, and healthcare systems. It promotes innovation, ethical practices, and interdisciplinary knowledge to enhance patient care and healthcare efficiency.

Course Objectives:

1. To develop advanced understanding of AI techniques for healthcare applications
2. To analyze complex clinical datasets using ML and DL models
3. To design AI-driven decision support systems for medical use cases
4. To evaluate ethical, legal, and regulatory challenges in healthcare AI
5. To assess real-world deployment readiness of medical AI systems

Teaching-Learning Process (General Instructions)

1. Case-based learning
2. Demonstrations of AI tools in healthcare
3. Clinical scenario discussions
4. Assignments and quizzes
5. Mini-project / case study

MODULE – I	**9 Hours**

INTRODUCTION: Introduction to AI in Healthcare

Overview of AI in medicine, Role of AI in clinical workflows, Intelligent systems in clinical workflows, Smart vs traditional healthcare systems, Basic Case studies

Application: Case study on clinical labs, AI-based radiology tools detect diseases like cancer from medical images quickly and accurately.

MODULE – II	**9 Hours**

Healthcare Data & ML Basics

Types of healthcare data: Electronic Health Record, images, biosignals-Basic ML concepts: Supervised & Unsupervised learning- Simple ML pipeline- Basic evaluation metrics (accuracy, precision)

Application: Using ML to analyze EHR, medical images, and biosignals for disease prediction, diagnosis, and patient monitoring.

MODULE – III	**9 Hours**

Disease prediction & diagnosis-Medical imaging basics-Patient, monitoring systems-Hospital management systems

Application: Applying AI in disease prediction, medical imaging analysis, patient monitoring, sentiment of treatment, and efficient hospital management systems.

MODULE – IV	**9 Hours**

Modern Trends in Healthcare AI

Introduction to Deep Learning-AI chatbots & Virtual Assistants Wearables & remote monitoring-Advanced DL models

Application: Using deep learning, AI chatbots, and wearable devices for advanced diagnosis, virtual assistance, and continuous remote patient monitoring.

MODULE – V	**9 Hours**

Ethics, Governance & Responsible AI

Bias, fairness, and trust in medical AI, Privacy, security, and data protection, AI as Software as Medical Device (SaMD), Regulatory frame works (FDA, CDSCO, HIPPA, DPDP, EU AI Act)and governance frameworks, Validation, clinical trials, and deployment challenges.

Application: Applying ethical AI principles, regulatory standards, and validation processes to ensure safe, fair, and secure deployment of medical AI systems.

AI Integration:

- Hospitals integrate AI systems into workflows to assist doctors in diagnosis and patient monitoring.
- Integrating supervised and unsupervised learning models into healthcare systems through a structured ML pipeline for automated decision support.
- Embedding AI models into imaging tools, monitoring devices, and hospital systems for real-time analysis and decision support.
- Integrating advanced deep learning models and AI assistants into healthcare platforms and wearable technologies for real-time insights and support.
- Integrating compliant AI solutions (SaMD) into healthcare with adherence to frameworks like FDA, CDSCO, HIPAA, DPDP Act, and EU AI Act.

Authentic Intelligence:

- AI enhances human intelligence by supporting doctors with accurate insights, not replacing them.
- Ensuring reliable, ethical, and interpretable AI by using accurate data, transparent models, and meaningful evaluation metrics like accuracy and precision.
- Ensuring accurate, trustworthy, and ethical outcomes through validated data, reliable models, and human oversight in healthcare processes.
- Ensuring ethical, transparent, and reliable AI by combining advanced models with human validation and secure, high-quality data handling.
- Ensuring trustworthy AI through bias mitigation, data privacy, rigorous clinical validation, transparent governance, and continuous monitoring after deployment.

Course Outcome:

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
CO1	Analyze AI concepts in clinical healthcare systems	L2
CO2	Apply ML/DL techniques to healthcare datasets	L4
CO3	Design AI-based solutions for medical applications	L3
CO4	Evaluate ethical and regulatory aspects of healthcare AI	L4
CO5	Assess deployment readiness of AI healthcare solutions	L5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	3	1	-	1	-	-	-	-	-	1	3	1
CO-2	3	3	2	2	3	-	-	1	-	-	2	3	3
CO-3	3	3	2	3	2	-	-	1	-	-	2	3	3
CO-4	3	3	3	2	3	-	1	2	1	1	2	3	3
CO-5	3	3	3	2	3	1	1	2	2	1	2	3	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS (TB):

1. S. Dash, B. Shickel, M. A. Bray, and S. Saria, Machine Learning for Healthcare, MIT Press, Cambridge, 2021
2. E. Topol, Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again, Basic Books, New York, 2019.
3. A. Holzinger, Explainable AI in Healthcare, Springer Nature, Cham, 2022.
4. K. D. Kulkarni and P. G. Bhat, Artificial Intelligence and Machine Learning in Healthcare, Academic Press (Elsevier), 2020.

REFERENCE BOOKS:

5. Rajkomar, A., Dean, J., & Kohane, I., Machine Learning in Medicine, Cambridge University Press, 2019.
6. J. Jiang, J. Zhi, A. Ni, & D. Shen, Advanced Methods and Tools for ECG Data Analysis, Springer, 2019 — *(for AI in signal processing)*
7. Biomedical Informatics: Computer Applications in Health Care and Biomedicine, Springer, 2018.
8. L. L. Popescu, Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical

Informatics, IGI Global, 2021.
9. K. Friston et al., Statistical Parametric Mapping: The Analysis of Functional Brain Images, Academic Press, 2011 — *(For neuroimaging AI)*
10. D. S. Char, N. Shah, and D. Magnus, Implementing Machine Learning in Healthcare: From Ethical Challenges to Legal Frameworks, Cambridge University Press, 2020.
11. G. Litjens et al., Deep Learning for Medical Image Analysis, Elsevier, 2017.

E-Resources:

- IEEE, springer, mdp, science direct
- https://ocw.mit.edu/courses/hst-921-information-technology-in-the-health-care-system-of-the-future-spring-2009/
- https://www.nlm.nih.gov/
- https://www.who.int/

https://ocw.mit.edu/search/?q=Health%20care

BUILDING AI APPLICATIONS WITH LARGE LANGUAGE MODELS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VI

Subject Code	: 24AM3615	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	45 (15 hrs for Project)
L–T–P–S	: 2–0–0–2	**CIE :** 100 Marks	

Course Vision :

This course envisions empowering students to design and deploy intelligent applications using Large Language Models (LLMs) by integrating theory with real-world implementation. It aims to develop expertise in modern generative AI systems that can understand, generate, and reason over human language. The course emphasizes practical problem-solving using cutting-edge tools, APIs, and frameworks. It also promotes ethical, responsible, and scalable AI application development. Ultimately, the vision is to create industry-ready engineers capable of building next-generation AI-driven solutions.

Course Learning Objectives:

This course will enable students to:

1. To understand the fundamentals of Large Language Models and transformer architectures.
2. To develop skills in prompt engineering and LLM interaction techniques.
3. To design and build real-world AI applications using LLM APIs and frameworks.
4. To apply fine-tuning, evaluation, and optimization strategies for LLMs.
5. To analyze ethical, security, and deployment challenges in AI applications.

Course Description : This course introduces the principles and practices of building AI applications using Large Language Models (LLMs). Students will explore transformer architectures, prompt engineering, fine-tuning, and deployment strategies. The course covers the full lifecycle of LLM-based systems—from model understanding to application development and scaling. It also includes hands-on experience with tools such as APIs, frameworks, and real-world case studies. Emphasis is placed on ethical AI, evaluation techniques, and industry use cases.LLMs are powerful deep learning models capable of generating and understanding human-like text, widely used in applications such as chatbots, summarization, and content generation

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Lecture Method:*** *Includes traditional lectures along with concept-driven discussions supported by interdisciplinary research examples from social and behavioral sciences*
2. ***Interactive Teaching:*** *Active learning through problem-solving sessions, debates, and guided discussions on real-world social and behavioral scenarios.*
3. ***Visual Learning:*** *Use of data visualizations, dashboards, case demonstrations, and short videos to explain behavioral patterns and social dynamics.*
4. ***Collaborative Learning:*** *Group-based analysis of social datasets, network structures, and behavioral case studies*
5. ***Critical Thinking:*** *Faculty will pose higher-order thinking questions related to ethics, fairness, bias, and interpretability in AI-driven social systems..*
6. ***Application-Oriented Learning:*** *Continuous emphasis on translating AI concepts into real-world social applications such as governance, education, healthcare, and public policy.*

UNIT – I	**08 Hours**

INTRODUCTION TO LLMS AND GENERATIVE AI Overview of LLM and Generative AI Evolution of AI, NLP, and Generative AI - Language Models: N-grams to Transformers - LLM Fundamentals: tokens, embeddings, context - Transformer architecture and self-attention - Overview of LLM capabilities and limitations **Case Study:** AI-based analysis of public behavior during large-scale social or societal events.
UNIT – II **08 Hours**
Machine Learning for Behavioral Analysis Machine learning paradigms for social and behavioral data analysis. Supervised and unsupervised learning for behavioral prediction. Feature extraction and representation of social signals. Time-series and sequential modeling of human activities. Anomaly detection and behavioral profiling. Introduction to affective computing and emotion recognition. Evaluation metrics for behavioral models and fairness-aware machine learning. **Case Study:** Predicting student engagement and dropout patterns using educational and interaction data.
UNIT – III **08 Hours**
Prompt Engineering and Interaction Design : Prompt design principles -Zero-shot, few-shot, chain-of-thought prompting -System prompts and role-based interaction - Prompt optimization techniques - Prompt evaluation and debugging
UNIT – IV **08 Hours**
Building Applications with LLM APIs Introduction to APIs (OpenAI, Hugging Face, Gemini) - Building chatbots and conversational agents -Text generation, summarization, classification -Retrieval-Augmented Generation (RAG) -Integrating LLMs into web and mobile applications Fine-tuning vs prompt tuning - Transfer learning and domain adaptation Model evaluation metrics (accuracy, BLEU, ROUGE) - Performance optimization and cost management Scaling laws and inference optimization
UNIT – V **07 Hours**
Ethics, Policy, and Advanced Applications Ethical challenges in AI-driven social systems. Algorithmic bias and fairness in behavioral models. Privacy, surveillance, and data protection issues. Explainable and accountable AI for social decision-making. Applications of AI in governance, education, healthcare, and mental health. Behavioral nudging, persuasive AI, and social impact assessment. Regulatory frameworks and future directions in responsible and human-centric AI. **Case Study:** Ethical evaluation of AI-based predictive policing and social risk assessment systems.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain AI principles and architectures used in social and behavioral science applications.	L2
2	Analyze human behavior, social interactions, and decision-making processes using AI-driven models	L3
3	Apply machine learning, natural language processing, and network analysis techniques for social and behavioral data analysis.	L3
4	Examine social dynamics, group behavior, and collective decision-making using computational and AI-based approaches.	L4
5	Evaluate real-world AI-driven social systems with respect to fairness, privacy, ethical impact, and societal outcomes.	L4

Mapping Levels of COs to POs / PSOs														
COs				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2			2							2	2	2
CO2	3	3		2	2							2	2	2
CO3	3	3	2	2	3							2	2	2
CO4	3	3	3	2	3							2	2	2
CO5	3	2	2	2	3	2	2	2				3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration:

1. Integration of LLM APIs (OpenAI, Hugging Face) into applications.
2. Use of AI frameworks like LangChain for workflow orchestration.
3. Deployment of AI models in cloud environments (AWS, Azure, GCP).
4. Real-time AI systems such as chatbots, virtual assistants, and recommendation engines.
5. Integration with databases using Retrieval-Augmented Generation (RAG)

AI Authentication :

1. Model output verification using human-in-the-loop validation.
2. Bias detection and fairness evaluation in generated outputs.
3. Secure API usage with authentication keys and access control.
4. Detection of hallucinations and misinformation in LLM responses.
5. Ethical compliance and responsible AI governance practices

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Russell, S., & Norvig, P. Artificial Intelligence: A Modern Approach. Pearson, 2021.
2. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
3. Generative AI with Large Language Models" – DeepLearning.AI (2024 Edition)
4. Natural Language Processing with Transformers" – Lewis Tunstall et al. (O'Reilly, 2023)
5. Hands-On Large Language Models" – Jay Alammar & Maarten Grootendorst (2024)
6. Building LLM Powered Applications" – Chip Huyen (2024)
7. Aggarwal, C. C. Social Network Data Analytics. Springer, 2022.
8. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.

REFERENCE BOOKS:

1. Mitchell, T. M. Machine Learning. McGraw-Hill, 2017.
2. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.
3. Aggarwal, C. C. Machine Learning for Text. Springer, 2018.
4. Barocas, S., Hardt, M., & Narayanan, A. Fairness and Machine Learning. MIT Press, 2023.
5. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
6. Pentland, A. Social Physics: How Social Networks Can Make Us Smarter. Penguin Books, 2015.
7. Lazer, D. et al. Computational Social Science. Science, 2009.

E-Resources:

1. https://www.coursera.org/specializations/computational-social-science-ucdavis
2. https://www.elementsofai.com/

Activity Based Learning (Suggested Activities in Class)

1. Analysis of social and behavioral datasets using AI techniques through group discussion.
2. Case-study-based evaluation of AI models applied in governance, education, healthcare, or social media platforms.

****

VII SEM – ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	PROJ	24AM4701	Capstone Project Phase-I	0	0	0	8	100	40	100	04
2	PEC	24AM47XX	Professional Elective Course – IV	3	0	0	0	60	40	100	03
3	PEC	24AM47XX	Professional Elective Course – V/ MOOC	3	0	0	0	60	40	100	03
4	OEC	24OEXXXX	Open Elective – II	3	0	0	0	60	40	100	03
			Total	9	0	0	8				**13**

IPCC-Integrated Professional Core Course, PCC-Professional Core Courses, SEC-Skill Enhancement Courses, PEC-Professional Elective Courses, OEC-Open Elective Courses, PROJ-Project Work, L – Lecture, T – Tutorial, P – Practical, J–Project, C – No. Of Credits

Professional Elective Courses –IV (PEC-IV) Offering												
SL	**Course Type**	**Course Name**	**Teaching Department**	**Teaching Hours / Week**				**Examination**				**Credits**
				Lecture	**Tutorial**	**Practica**	**Project**	**Duration in Hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
				L	**T**	**P**	**J**					
1	PEC	Autonomous Planning and Reasoning Agents		2	0	0	2	03	100	--	100	**03**
2	PEC	Intelligent Video Surveillance Systems		2	0	0	2	03	100	--	100	**03**
3	PEC	Foundation Models for Robotics		2	0	0	2	03	100	--	100	**03**
4	PEC	AI for Agriculture		2	0	0	2	03	100	--	100	**03**
5	PEC	Multi modal and Multi lingual Language Perception		2	0	0	2	03	100	--	100	**03**
DLI Integrated												
5	DLI/MOOC	AI and Data Science										

Professional Elective Courses - V (PEC-V) Offering												
SL	**Course Type**	**Course Name**	**Teaching Department**	**Teaching Hours / Week**				**Examination**				**Credits**
				Lecture	**Tutorial**	**Practica**	**Project**	**Duration in Hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
				L	**T**	**P**	**J**					
1	PEC	Operationalizing Agentic AI Systems		2	0	0	2	03	100	--	100	**03**

2	PEC	Multi-modal interaction and speech interfaces		2	0	0	2	03	100	--	100	**03**
3	PEC	Human-Centric Automation and Industry 5.0		2	0	0	2	03	100	--	100	**03**
4	PEC	Blockchain for Cybersecurity and Data Privacy		2	0	0	2	03	100	--	100	**03**
5	PEC	Building Transformer-Based Natural Language Processing Applications		2	0	0	2	03	100	--	100	**03**

Open Elective Course (OEC-II) Offering												
SL	**Course Type**	**Course Name**	**Teaching Department**	**Teaching Hours / Week**				**Examination**				**Credits**
				Lecture	**Tutorial**	**Practica**	**Project**	**Duration in Hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
				L	**T**	**P**	**J**					
1	OEC	Responsible AI & Ethics		2	0	0	2	03	100	--	100	**03**
DLI Integrated												
2	DLI	GenAI and LLM										

CAPSTONE PROJECT PHASE - I

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 24AM4701	**Credits**	: 04
Hours / Week	: 04	**Total Hours**	: 42 Hours
L–T–P–J	: 0–0–0–8		

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To detail description** of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature.
7. **To analyze** and synthesize the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

7. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
8. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
9. Show Video/animation films to explain the functioning of various concepts.
10. Encourage Collaborative (Group Learning) Learning in the class.
11. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
12. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document for the chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team
   1. To be dedicated and committed to work on a new research topic by learning new technical skills
   2. To have fair knowledge on what is product development or research topic
   3. To have constant interaction with allocated guide by providing weekly updates
   4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
2. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics
3. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

b. Department staff as course guide

1. Ability to provide research direction to the student in the chosen field of interest
2. Ability to design an appropriate research strategy and methodology to carry out the research by student
3. Ability to provide and evaluate the strong literature review document for the chosen research topic
4. Ability to train students on research paper / technical writing skills
5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team

1. To be dedicated and committed to work on a new research topic by learning new technical skills
2. To have fair knowledge on what is product development or research topic
3. To have constant interaction with allocated guide by providing weekly updates
4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. Phase-1 comprises of Literature Survey, Problem identification, Objectives and Methodology.
2. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department based on the rubrics
3. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2

3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

AUTONOMOUS PLANNING AND REASONING AGENTS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER VII

Subject Code	: 24AM4702	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	45 (15 hrs for Project)
L–T–P–S	: 2–0–0–2	**CIE:** 100 Marks	

Course Learning Objectives:

This course will enable students to:

1. **Understand** the fundamental principles of classical planning, search algorithms, and heuristic design for automated problem solving.
2. **Analyze** and formulate sequential decision-making problems using Markov Decision Processes and related frameworks.
3. **Apply** dynamic programming, Monte Carlo methods, and temporal-difference learning techniques for value-based reinforcement learning.
4. **Integrate** planning and learning approaches for model-based reinforcement learning and online decision-making.
5. **Develop** policy gradient methods and actor-critic algorithms for continuous control and high-dimensional decision spaces.
6. **Evaluate** planning and decision-making systems with respect to optimality, sample efficiency, scalability, and real-world deployment considerations.

Teaching-Learning Process (General Instructions):

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Case Studies:*** *Use real-world applications from robotics, game playing, resource management, and autonomous systems to demonstrate the practical relevance of planning and decision-making algorithms.*
2. ***Problem-Solving and Design Exercises:*** *Conduct problem-solving sessions where students formulate planning domains, design heuristics, and analyze algorithmic trade-offs for sequential decision problems.*
3. ***Hands-on Learning****: Organize laboratory sessions involving implementation and evaluation of planning algorithms, MDPs, and reinforcement learning agents in simulated environments.*
4. ***Reflective and Research-Based Learning:*** *Encourage students to study recent research papers on planning and RL, reflect on algorithmic advances, limitations, and design choices through presentations and discussions.*

UNIT – I	**8 Hours**

Classical Planning & Search:

Planning problem formulation: States, actions, goals, and plans. STRIPS and PDDL representation languages. State-space search: Forward and backward search. Uninformed search: BFS, DFS, iterative deepening. Informed search and heuristics: A* algorithm, greedy best-first search. Heuristic design: Relaxed planning graphs, delete-relaxation, pattern databases. Planning graphs

and GraphPlan algorithm. Hierarchical Task Network (HTN) planning.

UNIT – II	**8 Hours**
Markov Decision Processes & Dynamic Programming: Sequential decision-making under uncertainty. Markov Decision Processes: States, actions, transition dynamics, rewards, policies. Bellman equations and optimality principles. Value iteration and policy iteration algorithms. Convergence properties and computational complexity. Partially Observable MDPs (POMDPs): Belief states and belief-space planning. Extensions: Discounting, finite vs infinite horizons, episodic vs continuing tasks.	
UNIT – III	**8 Hours**
Value-Based Reinforcement Learning: Model-free reinforcement learning framework. Monte Carlo methods: First-visit and every-visit MC, exploring starts. Temporal-Difference learning: TD(0), TD(λ), eligibility traces. Q-learning and SARSA algorithms. Function approximation in RL: Linear methods, feature engineering. Deep Q-Networks (DQN): Experience replay, target networks, double DQN. Rainbow and distributional RL. Multi-step bootstrapping and n-step methods.	
UNIT – IV	**7 Hours**
Policy Gradient & Actor-Critic Methods: Policy-based reinforcement learning. Policy gradient theorem and REINFORCE algorithm. Baseline functions and variance reduction. Actor-critic architectures: A2C, A3C. Advanced policy gradient methods: Proximal Policy Optimization (PPO), Trust Region Policy Optimization (TRPO). Deterministic policy gradients: DDPG, TD3. Soft Actor-Critic (SAC) and maximum entropy RL. Continuous action spaces and control problems.	
UNIT – V	**8 Hours**
Model-Based RL, Planning & Advanced Topics: Model-based reinforcement learning: Learning forward models, Dyna architecture. Monte Carlo Tree Search (MCTS) and AlphaGo/AlphaZero. Integrated planning and learning: Model Predictive Control with learned models. Multi-agent reinforcement learning: Nash equilibria, cooperative and competitive settings. Hierarchical reinforcement learning: Options framework, feudal networks. Exploration strategies: ε-greedy, UCB, Thompson sampling, curiosity-driven exploration. Sample efficiency, transfer learning, and sim-to-real transfer. Safety and robustness in planning and RL systems.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Analyze planning problems and design effective heuristics for state-space search algorithms.	L3
2	Formulate and solve sequential decision problems using MDPs and	L4

	dynamic programming methods.	
3	Implement and compare value-based reinforcement learning algorithms including deep Q-learning.	L3
4	Apply policy gradient and actor-critic methods to continuous control and high-dimensional problems.	L4
5	Critically evaluate model-based planning, exploration strategies, and deployment considerations in real-world RL systems.	L3

Mapping Levels of COs to POs / PSOs														
COs				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	2	2	2						2		3	
CO2	3	3	2	2	2						2		3	2
CO3	3	3	2	3	3						2		2	3
CO4	3	2	2	3	3						2		2	2
CO5	3	2	3	3	2	1	1	2	2		1		3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	**Weightage of Marks**
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, 2020.
2. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, 2nd Edition, 2018.
3. Malik Ghallab, Dana Nau, and Paolo Traverso, Automated Planning: Theory and Practice, 2004.

REFERENCE BOOKS:

Dimitri P. Bertsekas, Dynamic Programming and Optimal Control, 4th Edition, 2017.

Csaba Szepesvári, Algorithms for Reinforcement Learning, 2010, Morgan & Claypool.

Mykel J. Kochenderfer, Decision Making Under Uncertainty: Theory and Application, 2015, MIT Press.

E-Resources:

- NPTEL – Reinforcement Learning (IIT Madras)
  https://nptel.ac.in/courses/106106143
- OpenAI Spinning Up in Deep RL
  https://spinningup.openai.com/
- David Silver's RL Course (UCL/DeepMind)

https://www.youtube.com/playlist?list=PLqYmG7hTraZDMOYHWgPebj2MfCFzFObQ

- UC Berkeley CS188 – Introduction to AI
  https://inst.eecs.berkeley.edu/~cs188/
- Stanford CS221 – AI: Principles and Techniques
  https://stanford-cs221.github.io/
- DeepMind x UCL Deep RL Course
  https://www.deepmind.com/learning-resources/reinforcement-learning-lecture-series
- Gymnasium (OpenAI Gym) Documentation
  https://gymnasium.farama.org/
- Stable-Baselines3 RL Algorithms
  https://stable-baselines3.readthedocs.io/

Activity-Based Learning (Suggested Activities in Class):

Hands-on activity, Quiz, Seminar.

INTELLIGENT VIDEO SURVEILLANCE SYSTEMS

SEMESTER – VII

Subject Code	:	24AM4703	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S	:	2–0–0–2	**CIE:** 100 Marks		

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamental components, processes, and architectures involved in video analytics and surveillance systems.
2. Apply various techniques for foreground extraction, background modeling, segmentation, and object tracking in video data.
3. Familiarize with different classifiers and learning methods including neural networks, deep learning models, fuzzy logic, Bayesian methods, and HMMs used for video-based analysis.
4. Explore the role of video analytics in security applications such as human action recognition, behavioral analysis, crowd monitoring, and perimeter security.
5. Examine the use of video analytics in business intelligence, traffic monitoring, and advanced driver assistance systems for decision-making and real-world deployment.

Course Description:

Video Analysis and Surveillance focus on developing intelligent systems that can interpret and analyze video data for real-world applications. The course introduces core concepts such as video analytics pipelines, pre-processing, feature extraction (PCA, SIFT, HOG), segmentation, and object tracking. It also covers classification techniques ranging from classical machine learning to deep learning models like CNNs, RNNs, and LSTMs for spatio-temporal analysis, along with performance evaluation methods.

The course emphasizes practical applications in security, surveillance, traffic monitoring, and business intelligence, including object detection, behavior analysis, crowd monitoring, and anomaly detection. Students gain hands-on experience in designing video analytics systems while exploring advanced topics such as edge computing and AI-powered surveillance, with a focus on ethical and responsible use of these technologies.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain the fundamental components and processes of video analytics including pre-processing, feature extraction, and classification pipelines.	L2
2	Apply foreground extraction, segmentation, and tracking techniques to analyze video data effectively.	L3
3	Compare and use classification approaches such as neural networks, deep learning models, fuzzy logic, Bayesian methods, and HMMs for video-based analysis.	L3
4	Evaluate video analytics approaches to recommend suitable solutions for security, business intelligence, and traffic monitoring applications.	L3
5	Design and develop intelligent video surveillance systems addressing real-world challenges while considering ethical and privacy implications.	L3

Course Outcome:

Course Modules:	
UNIT – I: Video Analytic Components	**09 Hours**
Introduction to Video Analytics, Need for Video Analytics, Overview of video Analytics systems, Video analytics pipeline and architecture, Pre-processing Techniques, Feature space representation (PCA, FLD, SIFT, HOG), Classifier design for video data, Feature selection and dimensionality reduction. ***Applications:*** Surveillance and security systems, Traffic monitoring and accident detection, Facial recognition systems, Industrial quality inspection, Human activity recognition, Video search and retrieval.	
UNIT – II: Foreground Extraction & Segmentation	**09 Hours**
Background Modeling-Background estimation techniques, Averaging-based methods, Gaussian Mixture Model (GMM), Optical Flow-based approaches, Image Segmentation-Region-based segmentation, Region splitting and merging, Watershed segmentation, Morphological Operations-Erosion and Dilation, Advanced Tracking. ***Applications:*** Object detection and tracking in surveillance, Motion detection in video analytics, medical image segmentation, Autonomous vehicle navigation, Industrial defect detection, Crowd and activity monitoring.	
UNIT – III: Classifiers for Video Analysis	**09 Hours**
Classical Machine Learning Approaches, Bayesian classifier, Decision trees and ensemble methods, Fuzzy Classifier, Deep Learning Approaches, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs) for temporal modeling, LSTM networks, Probabilistic Models, Hidden Markov Models (HMM) based classifier, Conditional Random Fields (CRFs), Performance Evaluation. ***Applications:*** Action and activity recognition, Video-based anomaly detection, Speech and gesture recognition, Emotion recognition from video, Autonomous driving perception systems, Smart surveillance and behavior analysis.	
UNIT – IV: Video Analytics for Security Applications	**09 Hours**
Object Detection and Recognition, Human behavioral analysis, Gait recognition, Human action recognition, Pose estimation, Crowd Analysis, Crowd density estimation, anomaly detection in crowds, Security Systems, Perimeter security and intrusion detection, Smart city surveillance applications. ***Applications:*** Smart surveillance systems, Intrusion and perimeter security, Crowd monitoring and management, Biometric identification (gait and pose), Suspicious activity detection, Smart city safety applications.	
UNIT – V: Business Intelligence, Traffic Monitoring & Advanced Topics	**09 Hours**
Business Intelligence Applications, Video Analysis for retail analytics, Customer behaviour analysis, Queue management systems, Traffic Monitoring, Traffic flow analysis, Speed estimation, Traffic violation detection, Advanced Driver Assistance Systems (ADAS), Video-based rendering and scene reconstruction, Activity recognition and summarization, 3D video surveillance, Edge computing for surveillance, Future trends: AI-powered surveillance, federated learning. ***Applications:*** Retail analytics and customer behavior analysis, Queue management systems, Traffic monitoring and violation detection, ADAS in smart vehicles, 3D surveillance and scene reconstruction, Edge-based real-time video analytics.	

Mapping Levels of COs to POs / PSOs					
COs				**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	2	1							2	1
CO2	2	3	2	3	2							3	2
CO3	2	3	3	3	2				1	1	_	3	3
CO4	1	2	3	3	3				2	2	1	3	3
CO5	1	2	2	3	2	1	1	3	2	2	2	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - Module-1: AI-based video understanding systems, automated feature extraction tools, AI-driven video classification models.
  - Module-2: AI-based background subtraction models, motion detection algorithms, intelligent object segmentation tools.
  - Module-3: Deep learning models (CNN, RNN, LSTM) for video classification, AI-based activity recognition systems, probabilistic sequence modeling tools.
  - Module-4: AI-powered surveillance systems, human behavior analysis models, crowd monitoring and anomaly detection systems.
  - Module-5: AI-driven retail analytics, smart traffic monitoring systems, ADAS and edge-based video analytics solutions.

- **Authentic Intelligence Reflection:**
  - Module-1: Students analyze limitations of video analytics systems, challenges in feature extraction, and the need for human interpretation in complex scenarios.
  - Module-2: Students evaluate accuracy issues in segmentation and tracking, challenges in dynamic environments, and robustness of motion detection techniques.
  - Module-3: Students assess model performance, overfitting issues, and limitations of deep learning models in real-time video applications.

- Module-4: Students examine ethical concerns in surveillance, privacy issues, and biases in behavior and facial recognition systems.
- Module-5: Students evaluate reliability of AI in traffic and retail analytics, challenges in edge deployment, and ethical implications of large-scale video monitoring.

RESEARCH INTEGRATION TEACHING

1. Paper review of recent research in video analytics and AI
2. Case studies on real-world surveillance and intelligent systems
3. Mini presentations on emerging topics in deep learning and computer vision
4. AI-assisted research using tools for literature review and comparison
5. Critical analysis of research gaps, limitations, and ethical considerations

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
    a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
    b. **Continuous Assessment Tools (CAT – Six Components):** It may include: Surprise test (Computer-based), Quiz, Project based learning, Reflection Note on Guest Lecture, Industrial visit, E-course certification, Building models, Group discussion, Case study, Seminar, or Open Book Assignment.
    c. **AI-Powered Assignment (APA):** Use of AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: AI-Based Video Classification Analysis

Students prompt an AI tool:
"Classify objects or activities in a given video dataset."

Students must:
1. Analyze classification results
2. Identify misclassifications and limitations
3. Suggest improvements

Assessment focuses on human judgment.

Assignment 2: Video Segmentation & Motion Detection Evaluation

Students ask AI:
"Explain background subtraction and motion detection techniques."
Students must verify:
1. Accuracy of segmentation methods
2. Performance in dynamic environments
3. Limitations in real-world scenarios

Assignment 3: AI-Based Tracking & Debugging Challenge

AI generates object tracking output/code
Students must:

1. Identify tracking errors or inconsistencies
2. Debug or improve the method
   Explain the issues and corrections

Assignment 4: Feature Extraction & Visualization

Students use AI tools to:

1. Extract features (HOG, SIFT, etc.) from video frames
2. Visualize feature representations
3. Analyze effectiveness of features

Assignment 5: Video Analytics System Evaluation

Students prompt AI:
"Design a video surveillance system for anomaly detection."

Students must compare:
1. Accuracy vs computational cost
2. Real-time feasibility
3. Ethical and privacy concerns

Suggested Reading:

1. Patil, Lalit V., et al. – Video Analytics, Nirali Prakashan, 2023.
2. Tekalp, A. Murat – Digital Video Processing, Pearson, 2015.

E-Resources: (Recommended Tools)

- Jones, Graeme A., et al. – Video-Based Surveillance Systems: Computer Vision and Distributed Processing, Springer
- Dey, Nilanjan, et al. – Applied Video Processing in Surveillance and Monitoring Systems, IGI Global
- Shan, Caifeng, et al. – Intelligent Video Surveillance Systems, Wiley
- Programming: Python (OpenCV, basic video processing), MATLAB (optional)
- Visualization: Python libraries (Matplotlib, Seaborn), video visualization tools
- AI Tools: ChatGPT / AI assistants, computer vision libraries (OpenCV, TensorFlow, PyTorch)
- Platforms: Video analytics frameworks, annotation tools (CVAT, Labeling)

Activity Based Learning (Suggested Activities in Class)

1. Collaborative Activity is minor project development with a team of 4 students.

****

FOUNDATION MODELS FOR ROBOTICS
SEMESTER-VII
Professional Elective

Subject Code	:	24AM4704	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S	:	2–0–0–2	**CIE:** 100 Marks		

Course Learning Objectives:

Students will:

1. Understand the principles and capabilities of foundation models in AI systems
2. Analyze limitations of traditional robotics pipelines and the need for integrated intelligence
3. Apply language and vision models for robotic perception and interaction
4. Examine how foundation models enable reasoning and decision-making
5. Design robotic applications integrating multimodal AI models
6. Evaluate reliability, safety, and ethical implications of AI-driven robotic systems

Course Description :

To enable students to understand and apply foundation models (LLMs, VLMs, and Vision-Language-Action models) in robotic systems for perception, reasoning, and task-level decision-making, representing a paradigm shift from traditional modular pipelines to integrated intelligent systems.

This course emphasizes how foundation models transform robotic autonomy by enabling language understanding, cross-modal reasoning, and generalization across environments, moving beyond rigid perception–planning–control architectures toward flexible, intelligent robotic behavior.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	Analyze and describe the principles and role of foundation models in robotics	L4
CO2	Apply language and vision models for perception and interaction tasks	L3
CO3	Analyze and utilize foundation models for reasoning and decision-making in robotics	L4
CO4	Develop and integrate foundation models into robotic subsystems	L5
CO5	Design and implement a complete multimodal robotic system	L6

UNIT – I : Foundations of Robotics Paradigm Shift	**09 Hours**

Evolution of robotics systems, modular vs integrated architectures, introduction to foundation models (LLMs, VLMs, VLAs), representation learning, limitations of classical robotics Applications: Natural language-based robot control, generalization across environments
UNIT – II : Language and Vision Models for Robotics **09 Hours**
Vision-Language Models (VLMs), multimodal embeddings, grounding language in perception, semantic scene understanding, perception-language alignment, linking perception outputs to actionable representations for robotic tasks. Applications: Language-guided object detection, scene interpretation, robot perception enhancement.
UNIT – III : Reasoning and Decision-Making with Grounded Robotics Actions **09 Hours**
Task decomposition, instruction following, prompt-based planning, reasoning strategies, grounding reasoning into executable robotic actions, decision-making under uncertainty, action abstraction and task-level planning. Applications: High-level task planning with executable robot actions, instruction-driven embodied agents.
UNIT – IV : Robotics Applications using Foundation Models **09 Hours**
Overview of applications including manipulation using vision-language-action models, semantic navigation, human-robot interaction, learning from demonstration, action execution constraints, embodied decision-making, case studies (RT-2, SayCan, PaLM-E) Applications: Language-driven manipulation, context-aware navigation, assistive and service robotics
UNIT – V : System Integration, Evaluation, and Future Robotics Intelligence **09 Hours**
System architectures for AI-driven robotics, ROS integration with foundation models, deployment challenges (latency, compute), evaluation of foundation model performance in robotics, failure case analysis (hallucination, grounding errors, safety risks), future trends ***Applications:*** End-to-end robotic systems, cloud vs edge AI in robotics

Mapping Levels of COs to POs / PSOs					
COs				**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and World	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	3	1	1	1	-	1	-	-	-	3	3	1
CO2	3	2	2	2	3	-	1	1	1	-	2	3	3
CO3	3	3	2	2	2	-	2	1	1	-	3	3	2
CO4	2	2	3	2	3	-	1	2	1	1	2	2	3
CO5	2	3	3	2	3	-	1	2	1	1	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Use of pretrained AI models in robotics workflows, API-based model access

Module-2: Integration of VLMs with robotic perception pipelines, transforming sensor data into actionable representations.

Module-3: LLM-based planners, prompt engineering for robotics tasks, integration of reasoning outputs with robotic execution pipelines.

Module-4: Combining perception, reasoning, and action in embodied systems

Module-5: Model orchestration, system-level integration strategies

Authentic Intelligence Reflection:

Module-1: Model bias, lack of interpretability, over-reliance on pretrained systems

Module-2: Visual misinterpretation, dataset bias, perception uncertainty, and challenges in grounding perception to real-world actions.

Module-3: Hallucination risks, mismatch between reasoning and physical execution, reliability and safety concerns in embodied decision-making.

Module-4: Real-world uncertainty, safety in human environments, embodiment limitations

Module-5: Accountability, trust, and real-world failure consequences

Mini Project (Mandatory)

Students will design a robotics system utilizing foundation models.

Examples:

- Language-guided object manipulation in simulation
- Vision-language-based navigation system
- Instruction-following robotic assistant

Mandatory Requirement:

All projects must demonstrate grounding of model outputs into robotic actions (simulation or real-world).

Students must analyze:

- System design and architecture
- AI model integration
- Task reasoning capability
- Robustness and limitations
- Ethical and safety considerations

Evaluation Focus

- System Design Clarity
- Level of Autonomy Achieved
- Grounding Fidelity
- Robustness Under Uncertainty
- Failure Handling and Recovery

Recommended Tools

- Python
- OpenAI / HuggingFace APIs
- Integration with ROS-based pipelines (optional but strongly encouraged)
- Simulation tools (Gazebo, PyBullet, Isaac Sim)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance

understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

1. Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach, 4th Edition, Pearson, 2020
2. Goodfellow, I., Bengio, Y., Courville, A., Deep Learning, MIT Press, 2016
3. Thrun, S., Burgard, W., Fox, D., Probabilistic Robotics, MIT Press, 2005

Research Papers:

1. Recent survey papers on foundation models in robotics (e.g., embodied AI surveys)
2. Google RT-2: Vision-Language-Action Models
3. SayCan: Language to Robot Action Grounding
4. PaLM-E: Embodied Multimodal Models
5. Recent publications from ICRA, RSS, CoRL (Post-2020)

****

AI FOR AGRICULTURE

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	:	24AM4705	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S		2–0–0–2	**CIE**	:	100 Marks

Course Learning Objectives:

This course will enable students to:

1. Understand the role of Artificial Intelligence, Machine Learning, and Computer Vision in modern agriculture.
2. Apply AI techniques for crop monitoring, yield prediction, and precision farming.
3. Analyze agricultural data from sensors, drones, and weather systems for intelligent decision-making.
4. Develop AI-enabled smart farming solutions using IoT, robotics, and automation tools.
5. Evaluate sustainability, efficiency, and future opportunities of AI-driven agriculture systems.

Course Description:

AI in Agriculture introduces the application of Artificial Intelligence technologies in transforming traditional farming into smart and sustainable agriculture. The course covers machine learning models for crop prediction, computer vision for disease detection, IoT sensors for soil and weather monitoring, and robotics for automated farming operations.

The Course explore how AI supports precision agriculture, optimizes resources, improves productivity, and reduces environmental impact. Topics include predictive analytics, deep learning, drones, autonomous tractors, agricultural data mining, and future trends in smart farming ecosystems.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Explain fundamentals of AI, ML, and computer vision in agriculture	L2
2	Apply AI models for agricultural data analysis and prediction	L3
3	Analyze crop disease, pest, and soil-health detection systems	L4
4	Develop AI-based smart farming solutions using IoT and robotics	L3
5	Analyze sustainability and efficiency of AI-enabled agriculture systems	L4

Course Outcome:

Course Modules :

UNIT – I : Introduction to AI in Agriculture	**09 Hours**

Evolution of agriculture technology, challenges in conventional farming, need for AI in agriculture, overview of AI, ML and data analytics, smart farming concepts, digital agriculture, AI pipeline (data → model → decision). ***Applications:*** Smart irrigation, crop advisory systems, farm monitoring.

UNIT – II : Precision Agriculture & Computer Vision	**09 Hours**

Precision agriculture concepts, Site-Specific Crop Management (SSCM), Variable Rate Technology (VRT), computer vision basics, image acquisition, crop disease detection, pest identification, weed

recognition, crop maturity detection using CNN models ***Applications:*** Disease diagnosis, automated spraying, crop quality grading.
UNIT – III : AI, IoT & Robotics in Farming **09 Hours**
IoT sensors for soil moisture, pH, temperature and humidity, sensor networks in farms, drones for surveillance and spraying, autonomous tractors, harvesting robots, robotics in greenhouse farming, AI + IoT integration. ***Applications:*** Smart irrigation, autonomous farming machines, precision spraying.
UNIT – IV : Agricultural Data Analytics & Prediction **09 Hours**
Agricultural datasets, data preprocessing, classification, clustering, regression, yield prediction models, weather forecasting, crop recommendation systems, satellite image analytics, decision support systems. ***Applications***: Yield estimation, crop planning, market forecasting.
UNIT – V : Sustainability, Future Trends & Smart Ecosystems **09 Hours**
Climate-smart agriculture, sustainable farming practices, AI for water management, carbon-efficient farming, blockchain in agriculture, future of agri-tech startups, ethical issues, challenges in rural deployment, future intelligent farming ecosystems. ***Applications:*** Smart villages, climate-resilient farming, agri-business intelligence.

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and the world	Ethics	Individual & Collaborative Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	1	-	2	-	1	-	-	-	2	1	1
CO2	3	3	2	1	3	-	-	-	-	-	2	2	2
CO3	3	3	3	2	3	-	-	-	-	-	2	2	2
CO4	3	3	3	2	3	1	1	2	1	-	2	2	3
CO5	3	3	2	2	2	2	3	1	1	1	3	2	2

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI INTEGRATION & AUTHENTIC INTELLIGENCE REFLECTION:

- **AI Integration:**
  - Module 1: AI-based farm decision systems

- Module 2: Deep learning for crop disease detection
- Module 3: IoT and robotics in agriculture
- Module-4: ML for crop yield and weather prediction
- Module-5: Sustainable AI farming ecosystems.

- **Authentic Intelligence Reflection:**
  - Module-1: Evaluate reliability of AI crop prediction models
  - Module-2: Identify bias in agricultural datasets
  - Module-3: Analyze ethical impact of farm automation
  - Module-4: Reflect on sustainability of AI farming tools
  - Module-5: Assess rural adoption challenges.

RESEARCH INTEGRATION TEACHING

1. Case study review on smart farming startups.
2. Paper review on crop disease detection datasets.
3. Seminar on AI in Indian agriculture.
4. Mini presentations on drones and robotics.
5. Critical analysis of sustainable agriculture systems.

ASSESSMENT SCHEME:

1. **"Internal Assessment (IA)"** refers to the Continuous Internal Assessment (CIA) with a weightage of 60%. It includes:
   a. **Mid Semester Examination (MSE):** One descriptive examination to evaluate conceptual understanding of AI fundamentals and problem-solving approaches.
   b. **Continuous Assessment Tools (CAT – Six Components):** Includes activities such as quizzes, surprise tests, case studies, group discussions, project-based learning, seminars, industrial visits, reflection notes, model building, e-course certifications, or open book assignments.
   c. **AI-Powered Assignment (APA):** Students utilize AI tools for solving problems while demonstrating Authentic Intelligence through critical evaluation, validation of AI outputs, ethical reasoning, and responsible decision-making.
2. **"External Assessment (EA)"** refers to the Semester End Examination (SEE) with a weightage of 40%. This examination evaluates students' overall understanding of AI concepts, application skills, and their ability to analyze and solve problems effectively.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI-POWERED ASSIGNMENTS:

AI^2 Pedagogy: It is a structured teaching-learning approach where students use AI tools for problem-solving while simultaneously developing Authentic Intelligence through critical evaluation, ethical reasoning, and responsible decision-making.

Assignment 1: Crop Disease Detection

Prompt AI tool to generate disease classification model using CNN. Students must verify dataset quality and accuracy.

Assignment 2: Yield Prediction Study

Compare regression models for crop yield prediction.
Students evaluate performance metrics.

Assignment 3: Smart Irrigation Design

Use AI tool to generate IoT irrigation logic.
Students refine practical implementation.

Assignment 4: Drone Data Analytics

Use AI visualization tools for crop aerial data analysis.

Assignment 5: Sustainable Farming Evaluation

Compare AI farming solutions based on cost, water usage, and efficiency

Suggested Reading:

Primary Textbooks

1. Köse, U., Prasath, V. B., Mondal, M. R. H., Podder, P., & Bharati, S. (2024). *Artificial Intelligence and Smart Agriculture Technology*. Auerbach Publications. ISBN 978-1032289069.
2. Singh, R., Gehlot, A., Prajapat, M. K., & Singh, B. (2020). *Artificial Intelligence in Agriculture*. New India Publishing Agency. ISBN 978-9389547924.

Additional Resources

1. Russell, S. J., & Norvig, P. (2022). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson Education India. ISBN 978-9356063570.
2. Shelke, Suresh (2025). *AI for Agriculture*. Current Publications. ISBN 978-9349728219
3. IEEE research articles

E-Resources : (Recommended Tools)

- Python/OpenCV
- Tensorflow/ PyTorch
- Scikit-learn
- Kaggle Agricultural Datasets
- Drone Mapping Software

Activity Based Learning (Suggested Activities in Class)

1. Group discussion on AI applications in crop monitoring, irrigation, and disease detection.
2. Collaborative mini project in teams of 4 students such as smart irrigation, pest detection, or crop yield prediction system.

****

MULTIMODAL AND MULTILINGUAL LANGUAGE PERCEPTION

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 24AM4706	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	45(15 hrs for Project)
L–T–P–J	: 2–0–0–2		

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamentals of multimodal and multilingual language perception.
2. Analyze how language is processed across text, speech, vision, and audio modalities
3. Explore computational models for multilingual and cross-lingual understanding
4. Apply deep learning techniques for multimodal language perception tasks
5. Evaluate real-world applications involving multilingual and multimodal AI systems

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Multimodal AI systems used in translation, speech assistants, and vision-language tasks*
2. ***Problem-Solving Sessions:*** *Designing models for multilingual and multimodal perception*
3. ***Reflective Learning****: Discussions on bias, inclusivity, and ethical challenges in multilingual AI*

UNIT – I	**10 Hours**
Foundations of Multimodal and Multilingual Language Perception Introduction to human and machine language perception, multimodal communication, modalities in language perception (text, speech, vision, gesture), multilingualism and language diversity, challenges in multilingual AI, overview of multimodal AI systems, applications in real-world scenarios.	
UNIT – II	**10 Hours**
Multilingual Language Processing Multilingual NLP fundamentals, language typology and scripts, tokenization and representation challenges, cross-lingual word embeddings, multilingual language models, transfer learning across languages, low-resource languages, language identification and code-mixing.	
UNIT – III	**11 Hours**

Multimodal Representation Learning Multimodal data representations, feature extraction from text, speech, and images, joint embedding spaces, alignment of multimodal data, fusion techniques (early, late, hybrid fusion), attention mechanisms, transformers for multimodal perception.	
UNIT – IV	**10 Hours**
Multimodal and Multilingual Models Vision-language models, speech-text multimodal systems, multilingual speech recognition, multimodal machine translation, pretrained multimodal models, zero-shot and few-shot learning, bias and fairness in multimodal and multilingual models.	
UNIT – V	**11 Hours**
Applications, Evaluation, and Emerging Trends Multimodal sentiment analysis, emotion recognition, visual question answering, multilingual chatbots and assistants, accessibility technologies, evaluation metrics for multimodal and multilingual systems, ethical issues, privacy concerns, future research directions.	

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Understand fundamentals of multimodal and multilingual language perception	L2
2	Analyze multilingual language processing techniques	L3
3	Apply multimodal representation learning methods	L4
4	Evaluate multimodal and multilingual AI models	L5
5	Design applications using multimodal and multilingual perception systems	L4

Mapping Levels of COs to POs / PSOs														
				Program Outcomes (POs)									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
COs	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex	Modern Tool Usage	The Engineer and Society	Environment and Sustainabilit	Ethics	Individual & Team Work	Communication	Project management and Fina	Life-long Learning	Cognitive Outcome	Skill & Design Outcome

				Problems			y				nce			
CO 1	3	3	1			2						1	2	1
CO 2	3	3	2								1	1	3	2
CO 3	3	3	3	2	2					2	1	3	3	3
CO 4	3	2	2	3	3						1	2	2	2
CO 5	3	3	3	3	3						2	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H. *Speech and Language Processing*. Pearson, 3rd Edition, 2023.
2. Baltrusaitis, T., Ahuja, C., & Morency, L. "Multimodal Machine Learning." *IEEE Signal Processing Magazine*, 2019.
3. Eisenstein, J. *Natural Language Processing*. MIT Press, 2022.

REFERENCE BOOKS:

1. Goldberg, Y. *Neural Network Methods for Natural Language Processing*. Morgan & Claypool, 2017.
2. Poria, S., et al. Multimodal AI: From Theory to Practice. Springer, 2022.
3. Lewis, M., et al. "Multilingual Language Models." *ACL Proceedings*, 2020.

E-Resources:

- https://nptel.ac.in/courses/106105235
- https://ieeexplore.ieee.org/document/10953521/
- https://www.geeksforgeeks.org/MLP-tutorials/
- MLP - Course (nptel.ac.in)

Activity Based Learning (Suggested Activities in Class)
Hands on activity, Quiz, Seminar

OPERATIONALIZING AGENTIC AI SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER-VII

Course Code	:	24AM4707	**Credits**	:	03
Hours/Week	:	03	**Total Hours**	:	45(15 hrs for Projects)
L–T–P–J	:	2–0–0–2	**CIE**	:	100 Marks

Course Vision:

To enable learners to apply Artificial Intelligence techniques for understanding, analyzing, and influencing human behavior and social systems, fostering the development of responsible, ethical, and human-centered AI solutions for real-world societal challenges.

Course Objectives:

This course will enable students to:

1. Understand the fundamental principles of artificial intelligence as applied to social and behavioral science domains.
2. Analyze human behavior, social interactions, and decision-making processes using AI-driven models.
3. Apply machine learning, natural language processing, and network analysis techniques to real-world social and behavioral data.
4. Evaluate AI-based social systems with respect to fairness, ethics, privacy, interpretability, and societal impact.
5. Develop the ability to design responsible, human-centered AI solutions for behavioral prediction, social intervention, and policy support.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. **Lecture Method**: Includes structured lectures integrated with concept-driven discussions supported by contemporary research articles, global policy documents, and real-world case examples of agentic AI systems.
2. **Interactive Teaching**: Active learning through guided discussions, ethical dilemma analysis, and problem-solving sessions focused on safety, alignment, and decision-making challenges in autonomous AI agents.
3. **Visual Learning**: Use of documentaries, expert talks, system flow visualizations, and simulation videos to illustrate AI failures, governance mechanisms, and human–AI interaction dynamics.
4. **Collaborative Learning**: Group-based evaluation of AI governance frameworks, regulatory case studies, and comparative analysis of international AI policies and standards.
5. **Critical Thinking**: Faculty will pose higher-order questions encouraging students to reason about ethical trade-offs, accountability, risk mitigation, and long-term implications of agentic AI deployment.
6. **Application-Oriented Learning**: Continuous emphasis on applying ethical principles, safety strategies, and governance models to real-world agentic AI use cases across domains such as healthcare, transportation, and public systems.

UNIT–I | **09 Hours**

AI FOR SOCIAL AND BEHAVIORAL SCIENCE

Topics: Overview of social and behavioral sciences. Scope and objectives of computational social science. Role of artificial intelligence in understanding human behavior and social systems. Types of social and behavioral data: surveys, census data, digital traces, sensor data, and social media data. Human-centered AI and behavioral modeling. Differences between traditional statistical methods and AI-driven approaches. Challenges in social data analysis: bias, subjectivity,

uncertainty, causality, and interpretability.

Case Study: AI-based analysis of public behavior during large-scale social or societal events.

UNIT– II	**09 Hours**

AI for Behavioral Analysis

Topics: Machine learning paradigms for social and behavioral data analysis. Supervised and unsupervised learning for behavioral prediction. Feature extraction and representation of social signals. Time-series and sequential modeling of human activities. Anomaly detection and behavioral profiling. Introduction to affective computing and emotion recognition. Evaluation metrics for behavioral models and fairness-aware machine learning.

Case Study: Predicting student engagement and dropout patterns using educational and interaction data.

UNIT– III	**09 Hours**

Natural Language Processing for Social Intelligence

Topics: Text as a primary source of social data. Natural language processing pipeline for social and behavioral analysis. Sentiment analysis, emotion detection, and opinion mining. Stance detection and topic modeling. Discourse and narrative analysis. Detection of misinformation, fake news, and hate speech. Applications and limitations of large language models in social science research.

Case Study: Public opinion analysis using social media data during elections or public debates.

UNIT– IV	**09 Hours**

Social Network and Group Behavior Analytics

Topics: Fundamentals of social network theory. Graph representation of social systems. Network measures such as centrality, cohesion, and connectivity. Community detection and influence modeling. Information diffusion, rumor spreading, and social contagion. Homophily, echo chambers, and polarization. Multi-agent systems for modeling collective and group behavior.

Case Study: Modeling information diffusion and influence propagation on online social platforms.

UNIT– V	**09 Hours**

Agent AI Systems for Social Impact

Topics: Ethical challenges in AI-driven social systems. Algorithmic bias and fairness in behavioral models. Privacy, surveillance, and data protection issues. Explainable and accountable AI for social decision-making. Applications of AI in governance, education, healthcare, and mental health. Behavioral nudging, persuasive AI, and social impact assessment. Regulatory frameworks and future directions in responsible and human-centric AI.

Case Study: Ethical evaluation of AI-based predictive policing and social risk assessment systems.

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Explain AI principles and architectures used in social and behavioral science applications.	L2
2	Analyze human behavior, social interactions, and decision-making processes using AI-driven models	L3
3	Apply machine learning, natural language processing, and network analysis techniques for social and behavioral data analysis.	L3
4	Examine social dynamics, group behavior, and collective decision-making using computational and AI-based approaches.	L4

5	Evaluate real-world AI-driven social systems with respect to fairness, privacy, ethical impact, and societal outcomes.	L4

Mapping Levels of COs to POs/PSOs

Table: Mapping Levels of COs to POs/PSOs

COs	Program Outcomes (POs)											PSOs	
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	AI/ML System Design and Development	Ethical, Scalable, and Real-World AI Deployment
CO 1	3	2	–	1	1	–	–	–	–	–	–	2	1
CO 2	2	3	2	2	2	–	–	–	1	–	–	3	2
CO 3	2	3	3	2	3	–	–	–	2	1	–	3	2
CO 4	2	3	2	3	2	–	–	–	2	1	–	2	2
CO 5	2	3	2	3	2	2	1	2	2	2	1	2	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

PSO-1: Apply concepts of **Artificial Intelligence, Machine Learning, and data analytics** to model, analyze, and interpret social and behavioral data for real-world applications..

PSO-2: Evaluate and design **ethical, fair, and human-centered AI systems** addressing societal challenges with considerations for privacy, bias, and policy implications.

AI Integration:

UNIT 1: AI integration in this unit is achieved through analysis of real-world social datasets and AI-driven behavioral models using tools such as Python and Google Colab. Students explore social data sources like surveys and social media, applying AI techniques to understand human behavior and societal patterns. Case studies involving large-scale events help learners connect theoretical concepts with practical applications.

UNIT 2: This unit integrates AI through implementation of machine learning models for predicting and analyzing human behavior. Students use libraries such as Scikit-learn and TensorFlow to perform classification, clustering, and anomaly detection. Practical exercises include behavioral prediction and fairness-aware model evaluation using real datasets.

UNIT 3: AI integration focuses on analyzing textual data from social platforms using NLP techniques. Students implement sentiment analysis, opinion mining, and topic modeling using tools like NLTK and spaCy. They also explore large language model capabilities for social data interpretation, including detecting misinformation and hate speech.

UNIT 4: This unit integrates AI through graph-based modeling of social systems using tools such as NetworkX. Students analyze network structures, community detection, and information diffusion. Simulation of multi-agent interactions helps understand collective behavior and influence propagation in social environments.

UNIT 5: AI integration in this unit emphasizes evaluation and design of responsible AI systems. Students analyze ethical issues such as bias, fairness, and privacy using real-world case studies. Tools for explainability and fairness assessment are introduced, enabling learners to evaluate AI systems in domains like healthcare, governance, and public policy.

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Russell, S., & Norvig, P. Artificial Intelligence: A Modern Approach. Pearson, 2021.
2. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
3. Aggarwal, C. C. Social Network Data Analytics. Springer, 2022.
4. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.

REFERENCE BOOKS:

1. Mitchell, T. M. Machine Learning. McGraw-Hill, 2017.
2. Jurafsky, D., & Martin, J. Speech and Language Processing. Pearson, 2023.
3. Aggarwal, C. C. Machine Learning for Text. Springer, 2018.
4. Barocas, S., Hardt, M., & Narayanan, A. Fairness and Machine Learning. MIT Press, 2023.
5. Salganik, M. J. Bit by Bit: Social Research in the Digital Age. Princeton University Press, 2020.
6. Pentland, A. Social Physics: How Social Networks Can Make Us Smarter. Penguin Books, 2015.
7. Lazer, D. et al. Computational Social Science. Science, 2009.

E-RESOURCES:

1. NVIDIA AI Learning Resources – https://developer.nvidia.com/ai
2. Coursera AI & Data Science Courses – https://www.coursera.org
3. NPTEL AI Courses – https://nptel.ac.in
4. Stanford Online AI Programs – https://online.stanford.edu
5. Hugging Face NLP & LLM Resources – https://huggingface.co/learn

RECOMMENDED TOOLS

Python, Jupyter Notebook, Google Colab, Scikit-learn, TensorFlow, NLTK, spaCy, NetworkX.

MULTIMODAL INTERACTION AND SPEECH INTERFACES
[As per Choice Based Credit System (CBCS) scheme]
SEMESTER - VII

Course Code	:	24AM4708	**Credits**	:	03
Hours / Week	:	03	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–J	:	2–0–0–2	**CIE**	:	100 Marks

Course Vision:
This course develops an in-depth understanding of multimodal speech interfaces by integrating speech processing, machine learning, and human-computer interaction, enabling students to design intelligent systems that combine speech with visual, gesture, and contextual modalities.

Course Objectives:
Students will:

1. Understand principles of speech processing and multimodal interaction
2. Analyze speech-based and multimodal interface architectures
3. Design systems integrating speech with other modalities
4. Apply machine learning and deep learning in multimodal systems
5. Evaluate usability, ethics, and real-world applications of multimodal interfaces

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods, where teacher can use to accelerate the attainment of the various course outcomes:

1. Lecture method combined with interactive discussions
2. Brainstorming, group activities, and case-based learning
3. Demonstrations using audio-visual tools and datasets
4. Problem-solving sessions and analytical exercises
5. Encouraging critical thinking through real-world scenarios and design challenges

MODULE – I	**08 Hours**

Foundations of Multimodal Speech Interfaces

Topics:

Introduction to Human–Computer Interaction (HCI), evolution of multimodal interfaces, modalities (speech, text, vision, gesture, touch), characteristics of speech as a modality, multimodal interaction models, advantages and challenges of multimodal systems, overview of speech-enabled systems.

Applications: Voice assistants, smart devices, conversational agents, assistive technologies.

MODULE – II	**09 Hours**

Speech Signal Processing and Recognition

Topics: Speech production and perception, digital representation of speech signals, feature extraction techniques (MFCC, LPC, PLP), noise reduction and speech enhancement, speech datasets and corpora, automatic speech recognition (ASR) architecture, acoustic, language, and pronunciation models, HMM, GMM, DNN-based ASR, Text-to-Speech (TTS) systems, evaluation metrics (WER, MOS).

Applications: Speech recognition systems, voice-controlled interfaces, transcription systems.

MODULE – III	**06 Hours**

Multimodal Representation and Fusion

Topics: Multimodal data representation, feature extraction across modalities (speech, text, vision),

fusion strategies (early, late, hybrid), joint embedding spaces, alignment techniques, attention mechanisms, multimodal dialogue systems, context awareness, user modeling, ambiguity resolution. **Applications:** Multimodal assistants, emotion-aware systems, interactive AI systems.	
MODULE – IV	**11 Hours**
Machine Learning for Multimodal Interfaces **Topics:** Supervised and unsupervised learning for multimodal data, deep learning models (CNN, RNN, Transformers), attention-based multimodal fusion, transfer learning, multimodal embeddings, pre-trained models, real-time processing challenges, system optimization. **Applications:** Speech-enabled AI systems, intelligent interfaces, adaptive systems.	
MODULE – V	**11 Hours**
Applications, Evaluation, and Ethical Considerations **Topics:** Applications in healthcare, education, automotive, and smart environments, usability testing and user experience evaluation, accessibility in multimodal systems, privacy and security issues, bias in speech systems, ethical considerations, fairness and inclusivity, emerging trends in multimodal speech interfaces. **Applications:** Healthcare assistants, educational tools, autonomous systems, accessibility technologies..	

AI Integration:

AI-based speech recognition and synthesis systems
Deep learning models for multimodal fusion
Transformer-based architectures for speech and multimodal tasks
Pretrained speech and multimodal models
Evaluation using benchmark datasets and performance metrics

Authentic Intelligence:

Limitations of speech recognition in diverse environments
Bias in speech datasets and accent variability
Challenges in multimodal fusion and interpretation
Ethical concerns in voice-based systems (privacy, surveillance)
Reliability and trust in intelligent multimodal systems

Course Outcome

Course Outcomes (COs)	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Understand fundamentals of speech processing and multimodal interaction	L1
2	Analyze speech recognition and multimodal interface techniques	L2
3	Apply multimodal fusion and machine learning methods	L3
4	Evaluate multimodal speech systems using performance metrics	L4
5	Design and assess real-world multimodal speech applications	L3

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs

COs	Program Outcomes (POs)											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO-1	3	1	–	–	–	–	–	–	–	–	–	1	2	1
CO-2	3	3	2	2	2	–	–	–	–	–	–	–	2	2
CO-3	2	3	2	2	2	–	–	–	–	–	–	–	3	2
CO-4	2	3	2	2	3	–	–	–	1	–	–	–	3	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	2	2	3

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30

AI-powered Assignment “Or” Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Jurafsky, D., & Martin, J. H., *Speech and Language Processing*, Pearson
2. Oviatt, S. et al., *The Handbook of Multimodal–Multisensor Interfaces*, ACM

REFERENCE BOOKS:

1. Shneiderman, B. et al., *Designing the User Interface*, Pearson
2. Huang, X., Acero, A., & Hon, H., *Spoken Language Processing*, Prentice Hall

E-RESOURCES:

NPTEL courses on Speech Processing and AI
Lecture materials from IITs on Multimodal Interaction
Research papers from IEEE, ACM, and Springer

RECOMMENDED TOOLS

- Programming: Python
- Libraries: TensorFlow, PyTorch
- Speech Tools: Kaldi, Librosa
- NLP Tools: NLTK, SpaCy
- Visualization: Matplotlib

HUMAN-CENTRIC AUTOMATION AND INDUSTRY 5.0
SEMESTER-VII
Professional Elective

Subject Code	:	24AM4709	**Credits**	:	03
Hours / Week	:	03 Hours	**Total Hours**	:	45 (15 hrs for Project)
L–T–P–S	:	2–0–0–2			

Course Learning Objectives:

Students will:

1. Understand principles of human-centric automation and Industry 5.0
2. Analyze human-robot interaction and collaboration models
3. Apply safety, ergonomics, and human factors in system design
4. Design automation systems with humans in the loop
5. Evaluate real-world systems for usability, safety, and effectiveness

Course Description :

This course focuses on the design and deployment of human-centric automation systems with humans in the loop.

It complements autonomous and AI-driven robotics by addressing real-world deployment challenges involving human interaction, safety, and usability.

Students will develop the ability to design deployable, human-aware automation systems that integrate human capabilities into system-level decision-making.

Course Outcome	Description	Bloom's Taxonomy Level
CO1	**Analyze and describe the principles of Industry 5.0 and human-centric automation**	**L4**
CO2	**Analyze human-robot interaction mechanisms and interaction modeling approaches**	**L4**
CO3	**Apply safety, ergonomics, and human factors in system design**	**L3**
CO4	**Design human-in-the-loop automation systems**	**L6**
CO5	**Develop or evaluate real-world automation systems**	**L5**
UNIT – I : Industry 5.0 and Human-Centric Automation		**09 Hours**

Practical limitations of purely autonomous robotic systems in real-world deployment, limitations of full automation, role of humans in automation, human-centric design principles, introduction to collaborative systems Applications: Smart manufacturing, assistive robotics, semi-autonomous industrial systems

UNIT – II : Human-Robot Interaction (HRI)	**09 Hours**
Types of interaction (physical, cognitive), collaboration modes (coexistence, cooperation, collaboration), interaction modeling and human intent inference, interfaces (gesture, voice, wearable), shared autonomy. Applications: Collaborative robots (cobots), teleoperation, assistive robotic systems	

UNIT – III : Safety, Ergonomics, and Human Factors	**09 Hours**
Safety in Human-in-the-Loop systems, risk assessment, introduction to safety standards, ergonomics, usability and cognitive load, design trade-offs between safety, efficiency, and usability Applications: Industrial robot safety, exoskeletons, workspace design	

UNIT – IV : Human-in-the-Loop Automation Systems	**09 Hours**
Design and analysis of Human-in-the-Loop control architectures, system architectures with humans in the loop, decision sharing, supervisory control, adaptive automation, failure handling. Applications: Semi-autonomous vehicles, industrial supervision systems, AI-assisted control	

UNIT – V : Industrial Applications and Deployment	**09 Hours**
Human-centric manufacturing, service robotics, Overview of deployment challenges in real environments, system integration, validation and testing, evaluation metrics for human-centric systems, human performance vs system performance trade-offs, failure case analysis in Human-in-the-Loop systems. Applications: Industrial automation, healthcare robotics, logistics systems	

Mapping Levels of COs to POs / PSOs					
COs				**Program Outcomes (POs)**	**PSOs**

	1	2	3	4	5	6	7	8	9	10	11	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and World	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	2	2	-	-	-	2	2	-	-	-	2	2	-
CO2	2	3	2	2	-	2	2	2	-	-	2	2	2
CO3	2	3	3	2	-	3	3	-	-	-	2	2	3
CO4	2	3	3	2	2	3	2	2	2	1	3	3	3
CO5	2	3	3	3	2	3	3	2	2	-	3	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

AI Integration & Authentic Intelligence Reflection (AI^2 Pedagogy)

AI Integration:

Module-1: Adaptive automation, AI-assisted decision support

Module-2: Intent recognition, learning from demonstration

Module-3: Predictive safety systems, human monitoring

Module-4: Reinforcement learning with human feedback, adaptive policies

Module-5: Adaptive workflows, real-time decision systems

Authentic Intelligence Reflection:

Module-1: Trust in automation, ethical implications of human replacement vs augmentation

Module-2: Transparency, explainability, human trust

Module-3: Ethical responsibility in safety-critical systems, inclusivity

Module-4: Accountability, human override mechanisms

Module-5: Societal impact, sustainability, workforce considerations

Mini Project (Mandatory)

Projects must explicitly demonstrate Human-in-the-Loop interaction and its measurable impact on system performance.

Examples

- Collaborative robot workflow
- Human-in-the-loop control system

• Ergonomic assistive robotic system

Minimum Requirement (Mandatory for all)

• System Definition
• Human–System Interaction Design
• System Architecture
• Safety and Risk Analysis
• Ergonomics and Usability
• Implementation / Simulation
• Failure Handling
• Evaluation Metrics

Evaluation Focus

• Clarity of System Design
• Quality of Human Integration
• Safety and Risk Consideration
• Usability and Ergonomic Design
• Technical Implementation / Simulation
• Failure Handling and Robustness
• Real-World Relevance

Recommended Tools

• Simulation tools (Gazebo, Webots)
• ROS (optional)
• Human-in-the-loop simulation frameworks
• Visualization tool

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

AI Tool Usage for Teaching and Learning

AI tools will be used in this course as structured pedagogical support to enhance

understanding, system design, and problem-solving.

Instructor Use (AI as Co-Teacher)

- Generate alternative explanations, examples, and case studies
- Demonstrate system workflows, debugging strategies, and design variations

Student Use (AI as Teaching Assistant)

- Concept clarification and understanding
- Code assistance and debugging (with explanation)
- System design exploration and documentation support

Guidelines for Use

- All AI-assisted work must be understood and explainable by the student
- Blind use of AI-generated content is not acceptable
- Students must critically evaluate AI outputs and identify errors or limitations

Evaluation Integration

- Viva and project discussions will assess individual understanding
- Students may be asked to explain or modify AI-assisted solutions

Suggested Reading

1. European Commission – Industry 5.0: Towards a Sustainable, Human-Centric and Resilient European Industry, 2021
2. ISO 10218 & ISO/TS 15066 – Collaborative Robot Safety Standards
3. Goodrich & Schultz – Human-Robot Interaction: A Survey, Foundations and Trends in HCI
4. Recent IEEE / Springer / Elsevier papers on Human-Robot Interaction and Industry 5.0

****

BLOCKCHAIN FOR CYBERSECURITY AND DATA PRIVACY

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VII

Subject Code	: 24AM47010	**Credits**:	03
Hours / Week	: 03 Hours	**Total Hours**:	45(15 hrs for Projects)
L–T–P–S	: 2–0–0–2	**CIE:** 100	

Course Vision:

To equip learners with the knowledge of blockchain technology to enhance cybersecurity and ensure secure, transparent data privacy solutions.

To enable the design of decentralized systems that protect data integrity, prevent cyber threats, and build trust in digital environments.

Course Learning Objectives:

This course will enable students to:

1. Understand the fundamentals of blockchain technology and distributed ledger systems.
2. Analyze cryptographic mechanisms used in blockchain for secure data management
3. Explore blockchain-based solutions for cybersecurity challenges
4. Apply blockchain techniques to ensure data privacy, integrity, and access control
5. Evaluate real-world blockchain applications in secure systems and privacy-preserving architectures

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. ***Real-World Examples:*** *Incorporate real-world examples and case studies that demonstrate the relevance and application of blockchain deployments in cybersecurity, healthcare, and finance.*
2. ***Problem-Solving Sessions:*** *Organize problem-solving sessions where students can work through challenging problems together.*
3. ***Reflective Learning****: Encourage students to reflect on their problem-solving approaches and discuss the reasoning behind their solutions.*

UNIT – I	**9 Hours**

Introduction to Blockchain

Blockchain fundamentals, history and evolution, centralized vs decentralized systems, peer-to-peer networks, distributed ledger technology, block structure, blockchain types: public, private, consortium blockchains.

Application:

Using blockchain to enable secure, transparent transactions and decentralized data management across industries like finance, healthcare, and supply chains.

UNIT – II	**9 Hours**

Cryptography Foundations for Blockchain Security

Cryptographic hash functions, digital signatures, public key infrastructure (PKI), symmetric and asymmetric encryption, key management, cryptographic security assumptions in blockchain systems.

Application:

Using cryptographic techniques like hashing, encryption, and digital signatures to secure blockchain transactions and protect data integrity.

UNIT – III	**9 Hours**

Blockchain for Cybersecurity

Cybersecurity threats and attack models, blockchain-based identity management, secure authentication and authorization, distributed trust models, blockchain for intrusion detection systems, DDoS mitigation using blockchain

Application:

Using blockchain to enhance cybersecurity through decentralized identity management, secure authentication, and protection against threats like DDoS and intrusions.

UNIT – IV	**9 Hours**

Data Privacy and Privacy-Preserving Blockchain Techniques

Data privacy challenges in distributed systems, GDPR and regulatory compliance, privacy issues in public blockchains, anonymization and pseudonymization.

Application:

Applying privacy-preserving techniques in blockchain to protect sensitive data while ensuring compliance with regulations like GDPR.

UNIT – V	**9 Hours**

Applications, Smart Contracts, and Emerging Trends

Smart contracts and security issues, secure smart contract design, vulnerabilities and attacks, blockchain in healthcare data privacy, financial data security, supply chain security, IoT security using blockchain, scalability challenges, future trends and research directions in blockchain security.

Application:

Using smart contracts and blockchain to secure healthcare, finance, supply chains, and IoT systems while addressing vulnerabilities and scalability challenges.

AI Integration:

- Integrating AI with blockchain for intelligent data validation, fraud detection, and automated decision-making in decentralized systems.
- Integrating AI to enhance cryptographic security through anomaly detection, key management optimization, and threat prediction.
- Integrating AI with blockchain for intelligent threat detection, anomaly analysis, and automated response in distributed security systems.
- Integrating AI with blockchain to enhance data anonymization, detect privacy risks, and ensure secure handling of personal data in distributed systems.
- Integrating AI with blockchain to detect smart contract vulnerabilities, enhance security, and optimize performance in emerging decentralized applications.

Authentic Intelligence:

- Ensuring trust, transparency, and data integrity through decentralized consensus, immutable ledgers, and ethically aligned AI-driven processes.
- Ensuring trustworthy and secure systems by applying robust cryptographic principles, safeguarding keys, and maintaining transparency and integrity in data protection.
- Ensuring secure, transparent, and trustworthy cybersecurity solutions through decentralized trust, strong authentication, and ethical AI-driven monitoring.
- Ensuring ethical and trustworthy data usage through strong privacy mechanisms, regulatory compliance, and transparent handling of user information.
- Ensuring secure, reliable, and future-ready systems through robust smart contract design, continuous monitoring, and ethical, scalable blockchain innovations.

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Understand blockchain architecture and distributed ledger principles	L2
2	Analyze cryptographic techniques used in blockchain security	L3
3	Apply blockchain concepts to address cybersecurity threats	L4
4	Evaluate privacy-preserving mechanisms in blockchain systems	L5
5	Design secure and privacy-aware blockchain-based applications	L4

Mapping Levels of COs to POs / PSOs													
COs				**Program Outcomes (POs)**								**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**
CO1	3	3	1			2						2	1
CO2	3	3	2								1	3	2
CO3	3	3	3	2	2					2	1	3	3
CO4	3	2	2	3	3						1	2	2
CO5	3	3	3	3	3						2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS:

1. Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. *Bitcoin and Cryptocurrency Technologies*. Princeton University Press, 2023.
2. Antonopoulos, A. M., & Wood, G. *Mastering Ethereum*. O'Reilly Media, 2022.
3. Drescher, D. *Blockchain Basics*. Apress, 2021.

REFERENCE BOOKS:

1. Crosby, M., et al. "Blockchain Technology: Beyond Bitcoin." *Applied Innovation Review*, 2020.
2. Bashir, I. *Mastering Blockchain*. Packt Publishing, 2023.
3. Christidis, K., & Devetsikiotis, M. "Blockchains and Smart Contracts for the Internet of Things." *IEEE Access*, 2019.

E-Resources:

- https://nptel.ac.in/courses/106105235
- https://ieeexplore.ieee.org/document/10953521/
- https://www.geeksforgeeks.org/blockchain-tutorials/

blockchain - Course (nptel.ac.in)

****

VIII SEM- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

SL	Course Type	Course Code	Course Name	Teaching Hours / Week				Examination			
				Lecture	Tutorial	Practical	Project	CIE Marks	SEE Marks	Total Marks	Credit
				L	T	P	J				
1	PROJ	24AM4801	Capstone Project Phase-II	0	0	0	24	60	40	100	12
2	INT	24AM4802	Internship	0	0	0	6	100	--	100	03
			Total	0	0	0	30				**15**

PROJ-Project Work, INT-Internship, L – Lecture, T – Tutorial, P – Practical, J – Project, C – No. Of Credits

CAPSTONE PROJECT PHASE – II

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	: 24AM4801	**Credits**	: 12
Hours / Week	: 12	**Total Hours**	: 156 Hours
L–T–P–J	: 0–0–0–24		

Course Learning Objectives:

This course will enable students to:

1. **To identify** key research questions within a field to carry out research in a team.
2. **To identify** and summarize the literature review of the relevant field.
3. **To demonstrate** relevant referencing and inculcate new skills in various aspects of academic writing.
4. **To demonstrate** the knowledge and understanding of writing the publication/report.
5. **To showcase** the strong evidence on the clarity of the argument, understanding of the selected domain area and presentation of its technical information.
6. **To detail description** of the process of carrying out the independent research in written document along with results and conclusions with reference to the existing literature.
7. **To analyze** and synthesize the new research findings.

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The research topic proposed by both the guide and the student team should be approved by the department chairman to proceed further.
2. A degree of industrial input and involvement will be encouraged, and can be facilitated through existing academic-industrial collaborations or by addressing specific topics that are of interest to industrial partners.
3. The problem statement should be big enough to be carried out in two phases over the two semesters i.e., VII and VIII semesters in the VI year.
4. All projects will be closely supervised by the Project Guide with ongoing feedback and guidance at all stages of the project from conception to completion.
5. The following criteria will be checked by the department chairman to approve for theresearch proposal:

a. Department staff as course guide
   1. Ability to provide research direction to the student in the chosen field of interest
   2. Ability to design an appropriate research strategy and methodology to carry out the research by student
   3. Ability to provide and evaluate the strong literature review document for the chosen research topic
   4. Ability to train students on research paper / technical writing skills
   5. Conduct reviews in regular time period and submit the evaluation to department chairman

b. Student Team
   1. To be dedicated and committed to work on a new research topic by learning new technical skills
   2. To have fair knowledge on what is product development or research topic
   3. To have constant interaction with allocated guide by providing weekly updates
   4. To be committed to complete the project and submitting the technical paper within the stipulated time framed by the university

Evaluation:

1. The problem statement selected in Capstone Project Phase - I (VII semester) will be carried in the VIII semester.
2. Phase 2 comprises of the detailed design, implementation, and testing results during the internal and external review.
3. Each Project team needs to submit the technical paper or patent or participate in hackathons and project exhibitions as well as apply for various state and national funding agencies within the stipulated time frame by the university
4. There will be CIA evaluation (Project reviews) done by a committee of senior faculty of the Department.
5. Additionally, there will be a Semester end evaluation of the work done that would include an internal Faculty and an external academic expert.

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course, the student will be able to:		
1	Identify and Select an Appropriate Research Problem	L1
2	Explain and Summarize Relevant Literature	L2
3	Compare and Critically Analyze Relevant Research Papers	L3
4	Construct a Research Model and Perform Evaluation	L4
5	Create and Draft Publications or Demonstrations	L5

Mapping Levels of COs to POs / PSOs

Cos				Program Outcomes (POs)									PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

RESEARCH INTERNSHIP/ INDUSTRY INTERNSHIP/IN-HOUSE INTERNSHIP

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Subject Code	:	24AM4802	**Credits**	:	03
Hours / Week	:	-	**Total Hours**	:	78 Hours
L–T–P–S	:	0–0–0–6			

Course Learning Objectives:

This course will enable students to:

1. To expose students to the industrial environment
2. To create competent professionals for the industry.
3. To provide possible opportunities to learn, understand and sharpen the real time technical/managerial skills required at the job
4. To work on a problem assigned by a mentor at industry, prepare action plan and complete within time limit
5. To learn, create/prepare report for Project/research as used in industry with productive and efficient way
6. To strengthen industry-institute linkage and increase employability of the students

Teaching-Learning Process (General Instructions)

These are sample new pedagogical methods that teachers can use to accelerate the attainment of the various course outcomes.

1. Lecture method means it includes not only the traditional lecture method but a different type of teaching method that may be adopted to develop the course outcomes.
2. Interactive Teaching: Adopt Active learning that includes brainstorming, discussing, group work, focused listening, formulating questions, note-taking, annotating, and roleplaying.
3. Show Video/animation films to explain the functioning of various concepts.
4. Encourage Collaborative (Group Learning) Learning in the class.
5. To make Critical thinking, ask at least three Higher-order Thinking questions in the class.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.

COURSE CONTENT:

1. The course includes a 16 weeks of on-job training on current industry-relevant problem through supervised self-learning approach.
2. The internship is an individual activity.
3. The student should obtain approval from the chairman/supervisor to pursue.
4. A student shall submit a brief proposal about the work to be carried out in the internship, to a coordinator within 3 weeks, after starting the internship.
5. A comprehensive report is required to be prepared and submit to the department at the end of the semester.
6. A certificate shall be attached with this report duly signed by the competent authority of the industry for the successful completion of the internship.
7. An attendance report shall also be attached with this report.
8. The CIA evaluation will be done by faculty mentor or Industry Supervisor.
9. There is no SEE Exam for this course.

Course Outcome	**Description**	**Bloom's Taxonomy Level**
At the end of the course, the student will be able to:		
1	Comprehend the modern tools used in the field of AIML and engineering for product development.	L2
2	Demonstrate ethical conduct and professional accountability while working in a team for the benefit of society	L3
3	Analyze the resources requirement and planning to facilitate the project success	L4
4	Assess and adapt technical skills on industry environment	L5
5	Apply the modern industry practice for internship	L3

Mapping Levels of COs to POs / PSOs														
Cos				**Program Outcomes (POs)**									**PSOs**	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
	Engineering Knowledge	Problem Analysis	Design & Development	Conduct Investigations of Complex Problems	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual & Team Work	Communication	Project management and Finance	Life-long Learning	Cognitive Outcome	Skill & Design Outcome
CO1	3	2	2	1	1	1	1	1	2	2	3	2	2	1
CO2	2	2	3	2	2	1	1	1	2	2	3	2	1	2
CO3	3	3	3	2	3	2	2	1	2	2	2	2	2	2
CO4	2	3	2	3	2	1	2	2	2	2	2	3	2	2
CO5	1	1	2	1	2	1	1	1	3	3	2	2	3	3

3: Substantial (High) **2: Moderate (Medium)** **1: Poor (Low)**

****

Building Transformer-Based Natural Language Processing Applications

[As per Choice Based Credit System (CBCS) scheme]

SEMESTER – VIII

Course Code	:	24AM47011	**Credits**	:	3
Hours / Week	:	3	**Total Hours**	:	45(15 hrs for Project)
L–T–P-J	:	2–0–0-2	**CIE**	:	100 MARKS

Prerequisites:

Proficiency in a C programming language.

Course Objectives:

1. To provide a strong foundation in optimization concepts relevant to Natural Language Processing tasks.
2. To enable students to formulate NLP problems as mathematical optimization problems.
3. To study and apply classical and modern optimization algorithms used in NLP models.
4. To analyze optimization challenges in deep learning and transformer-based NLP architectures.
5. To develop the ability to select, implement, and evaluate efficient optimization strategies for large-scale NLP systems.

Teaching-Learning Process (General Instructions)

1. Classroom lectures using presentations and board work to explain theoretical concepts and mathematical foundations.
2. Proble m-solving sessions and numerical examples to strengthen understanding of optimization techniques in NLP.
3. Case studies and discussion of recent research papers related to optimization in modern NLP models.
4. Hands -on demonstrations, assignments, and mini-projects using Python-based NLP and deep learning frameworks.
5. Continuous assessment through quizzes, seminars, and student presentations to encourage self-learning and critical thinking.

MODULE– I	7 hours

Fundamentals of NLP and Optimization:Introduction to NLP tasks: text classification, sequence labeling, machine translation ,Mathematical formulation of NLP problems ,Review of linear algebra, probability, and information theory for NLP,Loss functions in NLP: cross-entropy, negative log-likelihood,Convex and non-convex optimization problems.

MODULE – II	7 hours

Classical Optimization Techniques for NLP: Gradient Descent and its variantsBatch-Stochastic, and Mini-batch Gradient Descent,First-order optimization methods,Second-order methods:

Newton's Method, Quasi-Newton (BFGS, L-BFGS)Optimization in traditional NLP models -Logistic Regression,Conditional Random Fields (CRF),Maximum Entropy models.	
MODULE – III	7 hours
Optimization in Deep Learning for NLP: Backpropagation and computational graphs,Vanishing and exploding gradients,Advanced optimizers: Momentum, Adagrad, RMSProp, Adam, AdamW,Regularization techniques: dropout, weight decay, early stopping,Optimization for word embeddings: Word2Vec, GloVe.	
MODULE – IV	7 hours
Optimization for Sequence Models and Transformers: Optimization in RNNs, LSTMs, and GRUs,Attention mechanisms and optimization aspects,Transformer architecture and training challenges,Learning rate scheduling and warm-up, strategies,Fine-tuning vs pre-training optimization.	
MODULE – V	7 hours
Advanced and Scalable Optimization Techniques: Large-scale and distributed optimization,Gradient clipping and normalization,Mixed precision training,Hyperparameter optimization methods,Case studies from real-world NLP systems.	

Course Outcomes:

Course Outcome	Description	Bloom's Taxonomy Level
At the end of the course the student will be able to:		
1	Formulate NLP problems as optimization tasks	C6
2	Apply classical optimization techniques to NLP models	C3
3	Implement and analyze optimization strategies in deep NLP models	C4
4	Optimize transformer-based language models	C5
5	Evaluate optimization methods for scalable NLP applications	C5

Mapping Levels of COs to POs / PSOs

Table: Mapping Levels of COs to POs / PSOs													
COS	Program Outcomes (POs)											PSOs	
	1	2	3	4	5	6	7	8	9	10	11	1	2
CO-1	3	1	–	–	–	–	–	–	–	–	–	1	2
CO-2	3	3	2	2	2	–	–	–	–	–	–	–	2
CO-3	2	3	2	2	2	–	–	–	–	–	–	–	3
CO-4	2	3	2	2	3	–	–	–	1	–	–	–	3
CO-5	1	2	2	3	3	–	–	–	2	1	–	2	2

3: Substantial (High) 2: Moderate (Medium) 1: Poor (Low)

Evaluation Scheme

Criteria	Weightage of Marks
Continuous Assessment Tool (CAT)	20
Mid-Semester Examination (MSE)	30
AI-powered Assignment "Or" Mini Project*	50
Total	**100 marks**

TEXT BOOKS (TB):

1 .Jurafsky, D., & Martin, J. H., *Speech and Language Processing*, 3rd Edition (Draft), Pearson.

2. Goodfellow, I., Bengio, Y., & Courville, A., *Deep Learning*, MIT Press.

REFERENCE BOOKS:

1. Boyd, S., & Vandenberghe, L., Convex Optimization, Cambridge University Press.
2. Goldberg, Y., Neural Network Methods for Natural Language Processing, Morgan & Claypool.
3. Murphy, K. P., Machine Learning: A Probabilistic Perspective, MIT Press.
4. Bottou, L., Curtis, F. E., & Nocedal, J., Optimization Methods for Large-Scale Machine Learning, SIAM Review.

E-Resources:

https://blog.truegeometry.com/tutorials/education/15970c4fdbca1fc48c5b412d94e798c0/JSON TO_ARTCL_Optimization_in_Natural_Language_Processing_in_context_of_Optimiza.html ?utm_sourc e=chatgpt.com

Activity Based Learning (Suggested Activities in Class)

1. Real-world problem solving using group discussion– Students analyze real NLP problems (text classification, sentiment analysis, machine translation) and discuss suitable optimization strategies.
2. Role play activity– Role play of optimizer components such as gradient descent, learning rate scheduler, loss function, and model parameters to understand optimization flow.
3. Demonstration of solution through programming– Instructor-led and student-driven demonstrations using Python (PyTorch/TensorFlow) to show optimization techniques like SGD, Adam, and learning rate tuning in NLP models.
4. Flipped classroom activity– Students study assigned research papers or video lectures on optimization methods (e.g., AdamW, transformer fine-tuning) and discuss insights in class.