R.M.D. ENGINEERING COLLEGE
(An Autonomous Institution)
R.S.M Nagar, Kavaraipettai, Gummidipoondi Taluk,
Thiruvallur District, Tamil Nadu- 601206
Affiliated to Anna University, Chennai / Approved by AICTE,
New Delhi/Accredited by NAAC / All the Eligible UG Programs are
accredited by NBA, New Delhi

B.TECH.- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
REGULATIONS – 2024
CHOICE BASED CREDIT SYSTEM

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Graduates will apply the knowledge of Artificial Intelligence and Machine Learning to work effectively in interdisciplinary environments and develop innovative solutions to real-world problems.

PEO2: Graduates will excel in professional careers or pursue higher education and research in Artificial Intelligence, Machine Learning, and related emerging technologies.

PEO3: Graduates will adapt to rapid technological advancements through lifelong learning, professional development, and continuous skill enhancement.

PEO4: Graduates will demonstrate ethical values, social responsibility, and leadership as engineers or entrepreneurs contributing to sustainable societal development.

PROGRAM OUTCOMES (POs)

After the successful completion of the program, the graduates will be able to:

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1: Graduates will be able to apply Artificial Intelligence and Machine Learning techniques using Centre of Excellence laboratories, faculty expertise, and industry-supported platforms to develop intelligent and automated solutions for real-world problems, excel in hackathons, and attain global competency through international certifications and professional excellence.

PSO2: Graduates will be able to design and deploy AI models using emerging technologies such as Generative AI, Large Language Models, and Quantum Computing through research, innovation, and the development of multidisciplinary projects.

MAPPING OF PROGRAMME EDUCATIONAL OBJECTIVES(PEOs) WITH PROGRAMME OUTCOMES(POs)

A broad relation between the programme objective and the outcomes is given in the following table

PROGRAMME EDUCATIONAL OBJECTIVES	PROGRAMME OUTCOMES										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
I	3	3	3	2	2	2	1	1	1	1	2
II	1	2	2	2	2	1	1	3	3	3	3
III	1	1	2	1	1	2	3	2	2	1	1
IV	2	3	3	3	2	1	1	2	2	3	3

MAPPING OF PROGRAM SPECIFIC OBJECTIVES WITH PROGRAMME OUTCOMES

A broad relation between the Program Specific Objectives and the outcomes is given in the following table

PROGRAMME SPECIFIC OBJECTIVES	PROGRAMME OUTCOMES										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
I	3	3	3	3	3	2	2	1	1	2	2
II	3	3	3	3	3	2	1	1	3	3	3

Contribution **1: Reasonable** **2: Significant** **3: Strong**

R.M.D. ENGINEERING COLLEGE
(An Autonomous Institution)
R.S.M Nagar, Kavaraipettai, Gummidipoondi Taluk, Thiruvallur District, Tamil Nadu- 601206 Affiliated to Anna University, Chennai / Approved by AICTE, New Delhi/Accredited by NAAC / All the Eligible UG Programs are accredited by NBA, New Delhi

B.E. / B.TECH- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

REGULATIONS – 2024

CHOICE BASED CREDIT SYSTEM

I - VIII SEMESTER CURRICULUM AND SYLLABI
(For the Students admitted in the Academic Year 2024-2025)

SEMESTER – I								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1	24MA101	Matrices and Calculus	BSC	4	3	1	0	4
2	24GE101	Heritage of Tamils	HSMC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
3	24CS101	Programming in C++ (Lab Integrated)	ESC	6	3	0	3	4.5
4	24CS102	Software Development Practices (Lab Integrated)	ESC	6	3	0	3	4.5
5	24CH101	Engineering Chemistry (Lab Integrated)	BSC	5	3	0	2	4
6	24EC101	Digital Principles and System Design (Lab Integrated)	ESC	5	3	0	2	4
MANDATORY COURSES								
7	24MC101	Students Induction Program (Non Credit)	MC	3 Weeks				
8	24MC102	Programming in C (Non Credit)	MC	40 Periods				
EMPLOYABILITY ENHANCEMENT COURSES								
‘9	24HS111	Interpersonal skills, Psychometric Analysis and Career Development	EEC	1	1	0	0	1
10	24GE111	Idea Lab I (Non Credit)	EEC	1	0	0	1	0
			TOTAL	**29**	**17**	**1**	**11**	**23**

SEMESTER – II								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSE								
1	24GE201	Tamils and Technology	HSMC	1	1	0	0	1
THEORY COURSES WITH LABORATORY COMPONENT								
2	24MA201	Linear Algebra and Applications (Lab Integrated)	BSC	5	3	0	2	4
3	24CS201	Data Structures (Lab Integrated)	ESC	6	3	0	3	4.5
4	24CS202	Java Programming (Lab Integrated)	ESC	6	3	0	3	4.5
5	24PH201	Physics for Information Science (Lab Integrated)	BSC	5	3	0	2	4
6	24AM201	Introduction to Artificial Intelligence (Lab Integrated)	ESC	4	2	0	2	3
LABORATORY COURSE								
7	24GE211	Idea Lab II	EEC	2	0	0	2	1
EMPLOYABILITY ENHANCEMENT COURSES								
8	24HS211	Innovation and Creativity Skills Development	EEC	1	1	0	0	1
AUDIT COURSE								
9	24MC201	Yoga for Stress Management (Non Credit)	AC	1	0	0	1	0
MANDATORY COURSE								
10	24MC105	Environmental Science & Sustainability (Non Credit)	MC	2	2	0	0	0
			TOTAL	**33**	**18**	**0**	**15**	**23**

SEMESTER – III								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1.	24GE301	Universal Human Values 2: Understanding Harmony	HSMC	3	2	1	0	3
2.	24MA301	Discrete Mathematics	BSC	4	3	1	0	4
3.	24CS301	Computer Architecture	PCC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4.	24CS302	Advanced Java Programming (Lab Integrated)	PCC	6	3	0	3	4.5
5.	24CS303	Database Management Systems (Lab Integrated)	PCC	6	3	0	3	4.5
6.	24AM301	Artificial Intelligence and Decision Making (Lab Integrated)	PCC	4	2	0	2	3
EMPLOYABILITY ENHANCEMENT COURSES								
7.	24GE311	Product Development Lab - I	EEC	2	0	0	2	1
8.	24CS311	Aptitude and Coding Skills I	EEC	3	0	0	3	1.5
9.	24CS312	Internship/Seminar (1 Week)	EEC	1	0	0	1	0.5
MANDATORY COURSES								
10.		Indian Constitution (Non Credit)	MC	1	1	0	0	0
			TOTAL	**33**	**18**	**1**	**14**	**25**

SEMESTER – IV								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES WITH LABORATORY COMPONENT								
1.	24MA401	Probability and Statistics (Lab Integrated)	BSC	5	3	0	2	4
2.	24CS402	Design and Analysis of Algorithms (Lab Integrated)	PCC	5	3	0	2	4
3.	24IT402	Web Development Frameworks (Lab Integrated)	PCC	6	3	0	3	4.5
4.	24AM401	Machine Learning (Lab Integrated)	PCC	4	2	0	2	3
5.	24AM402	Neural Networks (Lab Integrated)	PCC	4	2	0	2	3
6.		Professional Elective I	PEC	4	2	0	2	3
EMPLOYABILITY ENHANCEMENT COURSES								
7.	24GE411	Product Development Lab -II	EEC	2	0	0	2	1
8.	24CS411	Aptitude and Coding Skills II	EEC	3	0	0	3	1.5
AUDIT COURSES								
9.		Value Education (Non Credit)	AC	1	1	0	0	0
			TOTAL	**34**	**16**	**0**	**18**	**24**

SEMESTER – V								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1.	24AM502	Generative AI	PCC	3	3	0	0	3
2.		Professional Elective II	PEC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
3.	24AM501	Deep Learning (Lab Integrated)	PCC	5	3	0	2	4
4.	24CS304	Operating Systems (Lab Integrated)	PCC	3	2	0	2	3
5.		Professional Elective III	PEC	4	2	0	2	3
LABORATORY COURSES								
6.	24HS511	Professional Communication – I (TOEFL)	HSMC	4	0	0	4	2
EMPLOYABILITY ENHANCEMENT COURSES								
7.	24CS511	Advanced Aptitude and Coding Skills I	EEC	3	0	0	3	1.5
8.	24AM511	Internship and Career Readiness/Seminar (2 Weeks)	EEC	2	0	0	2	1
9.	24GE511	Product Development Lab - III	EEC	2	0	0	2	1
MANDATORY COURSES								
10.	24MC511	Essence of Indian Traditional Knowledge (Non Credit)	MC	1	1	0	0	0
			TOTAL	**32**	**13**	**0**	**19**	**21.5**

SEMESTER – VI								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1.		Open Elective I	OEC	3	3	0	0	3
2.		Professional Elective IV	PEC	3	3	0	0	3
3.	24AM603	Automata Theory and Compiler Design	PCC	3	3	0	0	3
THEORY COURSES WITH LABORATORY COMPONENT								
4.		Design Thinking (Lab Integrated)	HSMC	3	1	0	2	2
5.	24AM602	Data Visualization (Lab Integrated)	PCC	4	2	0	2	3
6.	24AM601	Agentic AI (Lab Integrated)	PCC	5	3	0	2	4
LABORATORY COURSES								
7.	24HS611	Professional Communication – II (TOEFL)	HSMC	2	0	0	2	1
EMPLOYABILITY ENHANCEMENT COURSES								
8.	24CS611	Advanced Aptitude and Coding Skills II	EEC	3	0	0	3	1.5
9.	24GE611	Product Development Lab – IV	EEC	2	0	0	2	1
AUDIT COURSES								
10.	24AC611	Personality Development (Non Credit)	AC	2	2	0	0	0
			TOTAL	**29**	**16**	**0**	**13**	**21.5**

SEMESTER – VII								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
THEORY COURSES								
1.	24AM701	Computer Vision	PCC	3	3	0	0	3
2.		Professional Elective VI	PEC	3	3	0	0	3
3.		Open Elective II	OEC	3	3	0	0	3
4.	24AM702	Reinforcement Learning Techniques (Lab Integrated)	PCC	4	2	0	2	3
THEORY COURSES WITH LABORATORY COMPONENT								
5.		Professional Elective V	PEC	4	2	0	2	3
EMPLOYABILITY ENHANCEMENT COURSES								
6.	24CS711	Internship/Seminar (4 weeks)	EEC	4	0	0	4	2
			TOTAL	**24**	**13**	**0**	**8**	**17**

SEMESTER – VIII								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
EMPLOYABILITY ENHANCEMENT COURSES								
1	24AM811	Project Work	EEC	16	0	0	16	8
			TOTAL	**16**	**0**	**0**	**16**	**8**

TOTAL CREDITS : 163

CREDIT SUMMARY

S. No.	Subject Area	Credits Per Semester								Credit Total	Percentage
		I	II	III	IV	V	VI	VII	VIII		
1	**HSMC**	1	1	3	-	2	3	-	-	10	6.13%
2	**BSC**	8	8	4	4	-	-	-	-	24	14.72%
3	**ESC**	13	12	-	-	-	-	-	-	28	17.18%
4	**PCC**	-	-	15	14.5	10	10	6	-	52.5	32.21%
5	**PEC**	-	-	-	3	6	3	6	-	18	11.04%
6	**OEC**	-	-	-	-	-	3	3	-	6	3.68%
7	**EEC**	1	2	3	2.5	3.5	2.5	2	8	24.5	15.03%
8	**MC**			✓	✓	✓	✓				
	Total	**23**	**23**	**25**	**24**	**21.5**	**21.5**	**17**	**8**	**163**	

HSMC – Humanities and Social Sciences including Management courses; **BSC** – Basic Science Courses; **ESC** – Engineering Science Courses including workshop, drawing, basics of electrical/mechanical/computer etc.; **PCC** – Professional Core Courses; **PEC** – Professional Elective Courses relevant to chosen specialization/branch; **OEC** – Open Subjects–Electives from other technical and/or emerging subjects **EEC** – Project Work, Seminar and Internship in Industry or elsewhere

PROFESSIONAL ELECTIVES

PROFESSIONAL ELECTIVES								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
DATA SCIENCE AND ANALYTICS								
1.	24AM901	Foundations of Data Science	PEC	4	2	0	2	3
2.	24AM902	Text and Speech Analytics	PEC	3	3	0	0	3
3.	24AM903	No SQL Databases	PEC	4	2	0	2	3
4.	24AM904	Advanced Databases	PEC	4	2	0	2	3
5.	24AM905	Image and Video Analytics	PEC	3	3	0	0	3
6.	24AM906	Stream Processing and Analytics	PEC	3	3	0	0	3
7.	24AM907	Social Network Analytics	PEC	3	3	0	0	3
8.	24AM908	Cognitive Science and Analytics	PEC	3	3	0	0	3
9.	24AM909	Web Analytics	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
APPLIED AI								
1.	24AM910	AI in Agriculture and Precision Farming	PEC	3	3	0	0	3
2.	24AM911	AI in Block Chain	PEC	3	3	0	0	3

3.	24AM912	AI in Cyber Security	PEC	3	3	0	0	3
4.	24AM913	Ethical and Responsible AI	PEC	3	3	0	0	3
5.	24EC012	Industrial IoT Applications	PEC	3	3	0	0	3
6.	24AM915	Intelligent Robots	PEC	3	3	0	0	3
7.	24AM916	Conversational AI	PEC	3	3	0	0	3
8.	24AM917	Game Development and Programming	PEC	3	3	0	0	3
9.	24AM918	AI for Marketing	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
AI AND CLOUD								
1.	24CS951	Cloud Foundations	PEC	3	3	0	0	3
2.	24CS501	Distributed and Cloud Computing	PEC	3	3	0	0	3
3.	24CS912	Virtualization	PEC	3	3	0	0	3
4.	24AM959	MLOps	PEC	4	2	0	2	3
5.	24CS913	Data Engineering in Cloud	PEC	3	3	0	0	3
6.	24CS920	Cloud Services Management	PEC	3	3	0	0	3
7.	24AM945	Frameworks for Cloud Data Analytics	PEC	4	2	0	2	3
8.	24CS915	Machine Learning for NLP in Cloud	PEC	3	3	0	0	3
9.	24CS916	Microservice Architecture	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
HIGH PERFORMANCE COMPUTING								
1.	24AM919	Multi-Core Architecture and Programming	PEC	3	3	0	0	3
2.	24AM920	GPU Architectures and Programming	PEC	3	3	0	0	3
3.	24AM921	High Performance Computing	PEC	3	3	0	0	3
4.	24AM922	Quantum Computing	PEC	3	3	0	0	3
5.	24AM923	Fundamentals of Accelerated Computing	PEC	3	3	0	0	3
6.	24AM960	Nature Inspired Computing Techniques	PEC	3	3	0	0	3
7.	24AM924	Scalable Machine Learning	PEC	3	3	0	0	3
8.	24AM925	Optimization Methods in Machine Learning	PEC	3	3	0	0	3
9.	24AM926	Applied Accelerated Artificial Intelligence	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
COMPUTATIONAL INTELLIGENCE								
1.	24AM951	Computer Network and Security	PEC	3	3	0	0	3
2.	24CS929	Soft Computing	PEC	3	3	0	0	3
3.	24AM928	Applied AI and ML	PEC	3	3	0	0	3
4.	24AM930	Evolutionary and Swarm Intelligence	PEC	3	3	0	0	3
5.	24AM931	Affective Computing	PEC	3	3	0	0	3
6.	24AM932	Recommender Systems	PEC	3	3	0	0	3
7.	24AM933	Neuromorphic Computing	PEC	3	3	0	0	3
8.	24AM934	Graph Neural Networks	PE	3	3	0	0	3
9.	24AM935	Explainable and Trustworthy AI	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
INTELLIGENT HEALTHCARE								

1.	24AM936	AI and ML for Healthcare	PEC	4	2	0	2	3
2.	24AM937	Medical Image Analysis	PEC	3	3	0	0	3
3.	24AM938	Clinical Data Science	PEC	3	3	0	0	3
4.	24AM939	Deep Learning in Genomics and Life Sciences	PEC	3	3	0	0	3
5.	24AM940	Smart and Interactive Healthcare Technologies	PEC	3	3	0	0	3
6.	24AM941	Bio-Informatics	PEC	3	3	0	0	3
7.	24AM942	Computational Neuroscience	PEC	3	3	0	0	3
8.	24AM943	Computational Biology and Sequence Analysis	PEC	3	3	0	0	3
9.	24AM944	Applied ML for Life Sciences	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
GENERATIVE AI								
1.	24AM929	Large Language Models	PEC	4	2	0	2	3
2.	24AM946	Multimodal AI	PEC	3	3	0	0	3
3.	24AM952	Vector Databases	PEC	4	2	0	2	3
4.	24AM953	Prompt Engineering and LLM Optimization	PEC	4	2	0	2	3
5.	24AM954	Ethical and Responsible Generative AI	PEC	3	3	0	0	3
6.	24AM955	RAG and Knowledge Systems	PEC	3	3	0	0	3
7.	24AM956	Generative AI for Software Engineering	PEC	3	3	0	0	3
8.	24AM957	GenAI Systems Engineering and LLMOps	PEC	3	3	0	0	3
9.	24AM958	Human-Centered GenAI	PEC	3	3	0	0	3
10.	24AM812	Capstone Project	PEC	12	0	0	12	6
QUANTUM INTELLIGENCE AND COMPUTATIONAL SCIENCES (QI&CS)								
1.	24AM947	Quantum Mechanics and Quantum Circuits For Computing	PEC	3	3	0	0	3
2.	24AM948	Quantum Optimization and Quantum Machine Learning	PEC	3	3	0	0	3
3.	24AM949	Quantum Algorithms	PEC	3	3	0	0	3
4.	24AM950	Industry Certification - IBM Qiskit	PEC	-	-	-	-	3
5.	24AM812	Capstone Project	PEC	12	0	0	12	6

OPEN ELECTIVE – OFFERED TO OTHER DEPARTMENTS								
Sl. No.	**Course Code**	**Course Title**	**Category**	**Contact Periods**	**L**	**T**	**P**	**C**
1.	24AM001	Foundations of Deep Learning	OEC	3	3	0	0	3
2.	24AM002	Soft Computing	OEC	3	3	0	0	3
3.	24AM003	Prompt Engineering	OEC	3	3	0	0	3
4.	24AM004	Introduction to Reinforcement Learning	OEC	3	3	0	0	3
5.	24AM005	Introduction to Generative AI	OEC	3	3	0	0	3
6.	24AM006	Agentic AI for Engineers	OEC	3	3	0	0	3

R2024 CURRICULUM OF
B.TECH. (HONOURS) IN ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
WITH SPECIALIZATION IN ANY ELECTIVE VERTICAL

Additional 18 credits to be completed from the courses offered in a specific Professional Elective Vertical

R2024 CURRICULUM OF
B.TECH. (HONOURS) IN ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Additional 18 credits to be completed from the courses offered in any Professional Elective Vertical

R2024
B.TECH. ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING WITH MINOR DEGREE

SI. No	NAME OF THE MINOR DEGREE	OFFERRING DEPARTMENT
1.	Internet of Things	Electronics and Communication Engineering
2.	Advanced Web Development	Computer Science and Business Systems
3.	Fintech and Blockchain	Computer Science and Business Systems

R2024

MINOR DEGREE CURRICULUM OFFERED BY

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

(for other B.E. / B.Tech. Programmes)

MINOR DEGREE IN ARTIFICIAL INTELLIGENCE

SI. No	COURSE CODE	COURSE TITLE	CONTACT PERIODS	L	T	P	C
1.	24AM007	Data Science for Engineers	3	3	0	0	3
2.	24AM008	Machine Learning Algorithms	3	3	0	0	3
3.	24AM009	Deep Learning Techniques	3	3	0	0	3
4.	24AM010	Foundations of Large Language Models	3	3	0	0	3
5.	24AM813	Capstone Project	12	0	0	12	6

SEMESTER – I

24MA101	MATRICES AND CALCULUS (Common to B.E./B.Tech., all branches except CSBS)	L	T	P	C
		3	1	0	4

OBJECTIVES:

The Course will enable learners to:

- Explain the concepts of matrix algebra techniques.
- Understand various techniques to solve second and higher order differential equations.
- Demonstrate simple applications of functions of several variables and vector calculus.
- Comprehend the basic concepts of multiple integrals.
- Illustrate elementary ideas of vector calculus.

UNIT I	MATRICES	12

Eigenvalues and Eigenvectors of a real matrix – Properties of eigenvalues and eigenvectors – Statement and applications of Cayley-Hamilton Theorem – Diagonalization of matrices by orthogonal transformation (excluding similarity transformation) – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms.

UNIT II	DIFFERENTIAL EQUATIONS	12

Second and Higher order linear differential equations with constant coefficients - Method of variation of parameters – Homogenous equation of Euler's and Legendre's type – System of simultaneous linear differential equations with constant coefficients.

UNIT III	FUNCTIONS OF SEVERAL VARIABLES	12

Total derivative – Differentiation of implicit functions – Jacobian and properties – Taylor's series for functions of two variables – Maxima and minima of functions of two variables– Lagrange's method of undetermined multipliers.

UNIT IV	MULTIPLE INTEGRALS	12

Double integrals – Change of order of integration – Area enclosed by Cartesian Coordinates (excluding polar coordinates) – Triple integrals (excluding spherical and cylindrical coordinates) – Volume of solids (Cartesian Coordinates only).

UNIT V	VECTOR CALCULUS	12

Gradient, divergence and curl (excluding vector identities) – Directional derivative – Irrotational and Solenoidal vector fields – Green's theorem in a plane, Stoke's theorem and Gauss divergence theorem (Statement only) - Simple applications involving cubes and rectangular parallelopipeds.

TOTAL: 60 PERIODS

OUTCOMES:

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

CO1: Compute the matrix inverse and their higher powers.
CO2: Solve second and higher order differential equations.
CO3: Determine the maxima and minima of functions of two variables.
CO4: Determine the volume and surface area using multiple integrals.
CO5: Evaluate integrals using the concept of vector calculus.
CO6: Apply matrix algebra techniques to diagonalize the matrix.

TEXT BOOKS:

1. Erwin Kreyszig, "Advanced Engineering Mathematics", John Wiley and Sons, 10th Edition, New Delhi, 2016.
2. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, New Delhi, 44th Edition, 2014.

REFERENCES:

1. James Stewart, “Calculus: Early Transcendentals”, Cengage Learning, 7th Edition, New Delhi,2015.
2. M. K. Venkataraman, “Engineering Mathematics”, Volume I, 4th Edition, The National Publication Company, Chennai, 2003.
3. Sivaramakrishna Dass, C. Vijayakumari, “Engineering Mathematics”, Pearson Education India, 4th Edition 2019.
4. H. K. Dass, and Er. Rajnish Verma, “Higher Engineering Mathematics”, S. Chand PrivateLimited, 3rd Edition 2014.
5. B.V. Ramana, “Higher Engineering Mathematics”, Tata McGraw Hill Publishing Company,6th Edition, New Delhi, 2008.
6. NPTEL course on "Engineering Mathematics - I", by Prof. Jitendra Kumar, IIT Kharagpur:https://onlinecourses.nptel.ac.in/noc21_ma58/preview

24GE101	HERITAGE OF TAMILS (Common to All Branches)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Recognize Tamil literature and its significance in Tamil culture.
- Introduce the Tamils' rich artistic and cultural legacy.
- Familiarize the different types of folk and martial arts that are unique to Tamil Nadu.
- Acquaint the concept of Thinai in Tamil literature and culture.
- Comprehend the significance of Tamil in developing Indian culture.

UNIT I	LANGUAGE AND LITERATURE	3

Language Families in India - Dravidian Languages – Tamil as a Classical Language - Classical Literature in Tamil – Secular Nature of Sangam Literature – Distributive Justice in Sangam Literature - Management Principles in Thirukural - Tamil Epics and Impact of Buddhism & Jainism in Tamil Land - Bakthi Literature Azhwars and Nayanmars - Forms of minor Poetry – Development of Modern literature in Tamil - Contribution of Bharathiyar and Bharathidhasan.

UNIT II	HERITAGE - ROCK ART PAINTINGS TO MODERN ART – SCULPTURE	3

Hero stone to modern sculpture - Bronze icons - Tribes and their handicrafts - Art of temple car making -- Massive Terracotta sculptures, Village deities, Thiruvalluvar Statue at Kanyakumari, Making of musical instruments - Mridhangam, Parai, Veenai, Yazh and Nadhaswaram - Role of Temples in Social and Economic Life of Tamils.

UNIT III	FOLK AND MARTIAL ARTS	3

Therukoothu, Karagattam, Villu Pattu, Kaniyan Koothu, Oyillattam, Leatherpuppetry, Silambattam, Valari, Tiger dance - Sports and Games of Tamils.

UNIT IV	THINAI CONCEPT OF TAMILS	3

Flora and Fauna of Tamils & Aham and Puram Concept from Tholkappiyam and Sangam Literature - Aram Concept of Tamils - Education and Literacy during Sangam Age - Ancient Cities and Ports of Sangam Age - Export and Import during Sangam Age - Overseas Conquest of Cholas.

UNIT V	CONTRIBUTION OF TAMILS TO INDIAN NATIONAL MOVEMENT AND INDIAN CULTURE	3

Contribution of Tamils to Indian Freedom Struggle – The Cultural Influence of Tamils over the other parts of India – Self-Respect Movement – Role of Siddha Medicine in Indigenous Systems of Medicine – Inscriptions & Manuscripts – Print History of Tamil Books.

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: State the role of Tamil literature in shaping Tamil Cultural roots.

CO2: Express the cultural and religious significance of Tamil art and sculptures.

CO3: Identify and describe the techniques of folk and martial arts.

CO4: Classify the role of Thinai concept in Tamil culture and literature.

CO5: Compare the idea of cultural and intellectual contributions of Tamils.

TEXT BOOKS & REFERENCES:

1. தமிழக வரலாறு – மக்களும் பண்பாடும் – கே.கே. பிள்ளை (வெளியீடு: தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்).
2. கணினித் தமிழ் – முனைவர் இல. சுந்தரம். (விகடன் பிரசுரம்).
3. கீழடி – வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம் (தொல்லியல் துறை வெளியீடு)
4. பொருநை – ஆற்றங்கரை நாகரிகம். (தொல்லியல் துறை வெளியீடு)

5. Social Life of Tamils (Dr.K.K. Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
Social Life of the Tamils - The Classical Period (Dr.S .Singaravelu) (Published by: International Institute of Tamil Studies.
Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D.Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by: International Institute of Tamil Studies.)
Keeladi - 'Sangam City Civilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
). Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
2. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL) – Reference Book

24CS101	PROGRAMMING IN C++ (Common to All Branches)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- To learn problem solving and programming fundamentals.
- To gain knowledge on pointers and functions.
- To apply the principles of object orientated programming.
- To understand operator overloading, inheritance and polymorphism.
- To use the functionalities of I/O operations, files build C++ programs using exceptions.

UNIT I PROGRAMMING FUNDAMENTALS 9+9

Procedural Languages vs Object Oriented Languages - Types of computer programming languages - Genesis of C++ - Program Life Cycle -Structure of C++ program - Identifiers - Variables - Keywords - Number System -Binary Number System -Octal Number System- Decimal Number System -Hexadecimal Number System - Data types - Constants - Errors – Operators- Expressions - Type Conversions - Control-Flow Statements - Conditional Statements - Iterative Statements - Unconditional Control Statements - Arrays - One-Dimensional Arrays - Two-Dimensional Arrays - Multi -Dimensional Arrays - Strings - String Manipulation Functions - Array of Strings.

List of Exercise/Experiments:

1. Write C++ programs for the following:
    a. Find the sum of individual digits of a positive integer.
    b. Compute the GCD of two numbers.
    c. Find the roots of a number (Newton ‘s method)
2. Write C++ programs using arrays:
    a. Find the maximum of an array of numbers.
    b. Remove duplicates from an array of numbers.
    c. Print the numbers in an array after removing even numbers.
3. Write C++ programs using strings:
    a. Checking for palindrome.
    b. Count the occurrences of each character in a given word.

UNIT II POINTERS AND FUNCTIONS 9+9

Pointers - Pointer Variables - Pointer Operators & Expressions -Pointers with Arrays -Functions - Scope Rules -Function Arguments -return Statement - Function Variables - Storage Classes - Types of storage classes - Create Header Files - User-Defined Functions - Inline Functions - Function Overloading -Recursion - Namespaces.

List of Exercise/Experiments:

1. Generate salary slip of employees using structures and pointers. Create a structure Employee with the following members:EID, Ename, Designation, DOB, DOJ, Basic pay

Note that DOB and DOJ should be implemented using structure within structure.

2. Compute internal marks of students for five different subjects using structures and functions.

UNIT III CLASSES AND OBJECTS 9+9

Concepts of Object Oriented Programming – Benefits of OOP – Simple C++ program - Classes and Objects - Member functions - Nesting of member functions - Private member functions - Memory Allocation for Objects - Static Data Members - Static Member functions - Array of Objects - Objects as function arguments - Returning objects - friend functions – Const Member functions - Constructors – Destructors.

List of Exercise/Experiments:

1. Write a program Illustrating Class Declarations, Definition, and Accessing Class Members.

2. Program to illustrate default constructor, parameterized constructor and copy constructors.

Practice Questions & Scenario Based Questions:

1. Imagine you are working as a software engineer at a tech company. Your team is developing a mathematical software library that will be used in various applications across the company. One of the features that your team lead has asked you to implement is a function that calculates the number of trailing zeros in the factorial of a number.
   The team lead has emphasized the importance of encapsulation in your implementation.
2. Create a C++ class Calculator representing a simple calculator. The class should have the following attributes and methods:

Attributes: Two operands and an operation (+, -, *, /)

Methods: Perform the operation and return the result

Implement constructors to initialize the calculator with default values (0,0) and with specified values. Also, implement a destructor to perform any necessary cleanup.

UNIT IV	OPERATOR OVERLOADING, INHERITANCE AND POLYMORPHISM	9+9

Operator Overloading - Overloading Using Friend functions – Inheritance – Types of inheritance – Virtual Base Class - Abstract Class – Constructors in Derived Classes - member class: nesting of classes.

Pointer to objects – this pointer- Pointer to derived Class - Virtual functions – Pure Virtual Functions – Polymorphism.

List of Exercise/Experiments:

1. Write a Program to Demonstrate the i) Operator Overloading. ii) Function Overloading.
2. Write a Program to Demonstrate Friend Function and Friend Class.
3. Program to demonstrate inline functions.
4. Program for Overriding of member functions.
5. Write C++ programs that illustrate how the following forms of inheritance are supported:
   a) Single inheritance b) Multiple inheritance c) Multi level inheritance d) Hierarchical inheritance.

Practice Questions & Scenario Based Questions:

1. Joy is a software developer at a 3D modeling company. The company is developing a new software tool that will be used by architects and engineers to design and analyze 3D models of various structures. One of the features that her project manager has asked is to implement a function that calculates the volume of basic 3D shapes like cylinders and cuboids. The project manager has emphasized the importance of using function overloading in her implementation..
2. Imagine you are a software developer tasked with creating a utility program for a school that handles student scores. The school wants a simple program where teachers can enter the scores of students for a particular test, and the program will then provide the highest and lowest scores among them. This will help the teachers quickly identify the top performer and the student who might need extra help.

Your task is to write program that satisfies the above scenario using **inline function**

3. Develop a software system to manage part-time worker students at a university. These students have unique attributes such as their name, student ID, hourly wage, and hours worked per week. Your goal is to create a C++ program that models this system.

Ramu is a software developer at a company specializing in developing software solutions for geometric shapes. Recently, a client approached with a request to create a program to calculate the areas of rectangles and triangles.

UNIT V	I/O, FILES AND EXCEPTIONS	9+9

C++ Streams – Unformatted I/O - Formatted Console I/O – Opening and Closing File – File

modes - File pointers and their manipulations – Templates – Class Templates – Function Templates - Exception handling.

List of Exercise/Experiments:

1. Program to demonstrate pure virtual function implementation.
2. Count the number of account holders whose balance is less than the minimum balance using sequential access file.
3. Write a Program to Demonstrate the Catching of all Exceptions.

Practice Questions & Scenario Based Questions:

1. Develop a simple library management system. Create a base class Book representing a book in the library. The class should have attributes such as title, author, and ISBN (International Standard Book Number). Implement a virtual function displayDetails() to display information about the book.
2. A software developer working on a banking application. One of the requirements is to analyze account holders' balances based on a minimum balance threshold. The application should read account information from a sequential access file, where each line represents an account record in the format: account_holder_name, balance.
3. The financial company is developing a new software tool that will be used by financial analysts to perform various calculations. One of the features of that project is to implement is a function that performs division of two numbers.

Write a program that takes two integer inputs, numerator and denominator, from the user. Implement error handling to check if the denominator is zero. If the denominator is zero, display the message "Division by zero is not allowed!" using an exception. If the denominator is not zero, calculate the result of the division and display it.

4. Mini project.

TOTAL: 45 (L) + 45 (P) = 90 PERIODS

OUTCOMES:

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

CO1: Solve problems using basic constructs in C++.
CO2: Implement C++ programs using pointers and functions.
CO3: Apply object-oriented concepts and solve real world problems.
CO4: Develop C++ programs using operator overloading and polymorphism.
CO5: Implement C++ programs using Files and exceptions.
CO6: Develop applications using C++ concepts

TEXT BOOKS:

1. Herbert Schildt, “The Complete Reference C++”, 4th edition, MH, 2015.
2. E Balagurusamy,” Object Oriented Programming with C++”, 4th Edition, Tata McGraw-Hill Education, 2008.

REFERENCES:

1. Karl Beecher,” Computational Thinking: A beginner's guide to problem-solving and programming”, BCS Learning & Development Ltd, 2017. (Unit 1)
2. Nell Dale, Chip Weems, “Programming and Problem Solving with C++”, 5th Edition, Jones and Barklett Publishers, 2010.
3. John Hubbard, “Schaum's Outline of Programming with C++”, MH, 2016.
4. Yashavant P. Kanetkar, “Let us C++”, BPB Publications, 2020
5. ISRD Group, “Introduction to Object-oriented Programming and C++”, Tata McGraw-Hill Publishing Company Ltd., 2007.
6. D. S. Malik, “C++ Programming: From Problem Analysis to Program Design”, Third Edition, Thomson Course Technology, 2007.
7. Paul Deitel, Harvey Deitel, “C++ How to Program”, 10th Edition, Pearson Education Inc. 2017.

8. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0129720024067194883 7_shared/overview

LIST OF EQUIPMENTS:

1. Standalone desktops with C/C++ compiler (or) Server with C/C++ compiler.

24CS102	SOFTWARE DEVELOPMENT PRACTICES (Common to All Branches)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- To discuss the essence of agile development methods.
- To set up and create a GitHub repository.
- To create interactive websites using HTML
- To design interactive websites using CSS.
- To develop dynamic web page using Java script.

UNIT I	AGILE SOFTWARE DEVELOPMENT AND Git and GitHub	9+9

Software Engineering Practices – Waterfall Model - Agility – Agile Process – Extreme Programming - Agile Process Models – Adaptive Software Development – Scrum – Dynamic Systems Development Method – Crystal – Feature Driven Development – Lean Software Development – Agile Modeling – Agile Unified Process – Tool set for Agile Process.

Introduction to Git –Setting up a Git Repository - Recording Changes to the Repository - Viewing the Commit History - Undoing Things - Working with Remotes -Tagging - Git Aliases - Git Branching - Branches in a Nutshell - Basic Branching and Merging - Branch Management - Branching Workflows - Remote Branches - Rebasing.

Introduction to GitHub – Set up and Configuration - Contribution to Projects, Maintaining a Project – Scripting GitHub.

List of Exercise/Experiments:

1. Form a Team, Decide on a project:
    a) Create a repository in GitHub for the team.
    b) Choose and follow a Git workflow
        - Each team member can create a StudentName.txt file with contents about themselves and the team project
        - Each team member can create a branch, commit the file with a proper commit message and push the branch to remote GitHub repository.
        - Team members can now create a Pull request to merge the branch to master branch or main development branch.
        - The Pull request can have two reviewers, one peer team member and one faculty. Reviewers can give at least one comment for Pull Request updating.
        - Once pull request is reviewed and merged, the master or main development branch will have files created by all team members.
2. Create a web page with at least three links to different web pages. Each of the web pages is to be designed by a team member. Follow Git workflow, pull request and peer reviews.
3. Form a Team, Decide on a project:
    c) Create a repository in GitHub for the team.
    d) Choose and follow a Git workflow
        - Each team member can create a StudentName.txt file with contents about themselves and the team project
        - Each team member can create a branch, commit the file with a proper commit message and push the branch to remote GitHub repository.
        - Team members can now create a Pull request to merge the branch to master branch or main development branch.
        - The Pull request can have two reviewers, one peer team member and one faculty. Reviewers can give at least one comment for Pull Request

updation.

- Once pull request is reviewed and merged, the master or main development branch will have files created by all team members.

4. Create a web page with at least three links to different web pages. Each of the web pages is to be designed by a team member. Follow Git workflow, pull request and peer reviews.

UNIT II	HTML	9+9

Introduction – Web Basics – Multitier Application Architecture – Cline-Side Scripting versus Server-side Scripting – HTML5 – Headings – Linking – Images – Special Characters and Horizontal Rules – Lists – Tables – Forms – Internal Linking – meta Elements – Form input Types – input and datalist Elements – Page-Structure Elements.

List of Exercise/Experiments:

1. Create web pages using the following:
    - Tables and Lists
    - Image map
    - Forms and Form elements
    - Frames

UNIT III	CSS	15

Inline Styles – Embedded Style Sheets – Conflicting Styles – Linking External Style Sheets – Positioning Elements – Backgrounds – Element Dimensions – Box Model and Text Flow – Media Types and Media Queries – Drop-Down Menus – Text Shadows – Rounded Corners – Colour – Box Shadows – Linear Gradients – Radial Gradients – Multiple Background Images – Image Borders – Animations – Transitions and Transformations – Flexible Box Layout Module – Multicolumn Layout.

List of Exercise/Experiments:

1. Apply Cascading style sheets for the web pages created.

UNIT IV	JAVASCRIPT BASICS	9+9

Introduction to Scripting – Obtaining user input – Memory Concepts – Arithmetic – Decision Making: Equality and Relational Operators – JavaScript Control Statements – Functions – Program Modules – Programmer-defined functions – Scope rules – functions – Recursion – Arrays – Declaring and Allocating Arrays – References and Reference Parameters – Passing Arrays to Functions – Multidimensional arrays.

List of Exercise/Experiments:

1. Form Validation (Date, Email, User name, Password and Number validation) using JavaScript.

UNIT V	JAVASCRIPT OBJECTS	9+9

Objects – Math, String, and Date, Boolean and Number, document Object – Using JSON to Represent objects – DOM: Objects and Collections – Event Handling.

List of Exercise/Experiments:

Implement Event Handling in the web pages.

Mini Projects-Develop any one of the following web applications (not limited to one) using above technologies.

a. Online assessment system
b. Ticket reservation system
c. Online shopping

d. Student management system
e. Student result management system
f. Library management
g. Hospital management
h. Attendance management system
i. Examination automation system
j. Web based chat application

TOTAL: 45 (L) + 45 (P) = 90 PERIODS

OUTCOMES:

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

CO1: Understand basic software engineering practices effectively.

CO2: Apply version control using Git and GitHub, and manage code repositories proficiently.

CO3: Design web applications using HTML, CSS, and JavaScript.

CO4: Analyze problems and create solutions using CSS for better web page presentation and usability.

CO5: Develop interactive web pages using JavaScript with an event-handling mechanism.

CO6: Apply the technological changes and improve skills continuously.

TEXT BOOKS:

1. Roger S. Pressman, "Software Engineering: A Practitioner's Approach", McGraw Hill International Edition, Nineth Edition, 2020.
2. Scott Chacon, Ben Straub, "Pro GIT", Apress Publisher, 3rd Edition, 2014.
3. Deitel and Deitel and Nieto, "Internet and World Wide Web - How to Program", Pearson, 5th Edition, 2018.

REFERENCES:

1. Roman Pichler, "Agile Product Management with Scrum Creating Products that Customers Love", Pearson Education, 1 st Edition, 2010.
2. Jeffrey C and Jackson, "Web Technologies A Computer Science Perspective", Pearson Education, 2011.
3. Stephen Wynkoop and John Burke, "Running a Perfect Website", QUE, 2nd Edition, 1999.
4. Chris Bates, "Web Programming – Building Intranet Applications", 3rd Edition, Wiley Publications, 2009.
5. Gopalan N.P. and Akilandeswari J., "Web Technology", Second Edition, Prentice Hall of India, 2014.
6. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013094421427470336209 9_shared/overview

LIST OF EQUIPMENTS:

1. Systems with either Netbeans or Eclipse
2. Java/JSP/ISP Webserver/Apache
3. Tomcat / MySQL / Dreamweaver or
4. Equivalent/ Eclipse, WAMP/XAMP

24CH101	ENGINEERING CHEMISTRY (Common for B.E -CSD, CSE and B.Tech-AD, IT)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Gain a comprehensive knowledge on polymers utilized in various industrial sectors.
- Knowledge on the fundamental principles of energy storage devices.
- Gain insights into the basic concepts and applications of chemical sensors and cheminformatics.
- Identify the different types of smart materials and explore their applications in Engineering and Technology.
- Assimilate the preparation, properties and applications of nanomaterials in various fields.

UNIT I INDUSTRIAL POLYMERS — 15

Polymers: Terminology – functionality, degree of polymerization, properties – glass transition temperature and molecular weight (weight average method).

Engineering plastics: preparation, properties and application of Polyethylene, Teflon, Polyethylene terephthalate and Epoxy resin- industrial applications of Polyethylene and Polyethylene terephthalate in packaging.

Special polymers: preparation, properties and applications of piezoelectric polymer - Polyvinylidene fluoride, electroactive polymer-Polyacetylene and biodegradable polymer - Polylactic acid.

(Theory-9)

1. Determination of the molecular weight of polymer using viscometer.
2. Determination of degradation of biodegradable polymer using photoreactor.
3. Demonstrate the applications of biodegradable plastic using 3D printing.

(Laboratory-6)

UNIT II ENERGY STORAGE DEVICES AND GREEN FUEL — 15

Introduction to Electrochemical cell and its terminology - electrochemical series and its applications.

Batteries: classification - construction and working principle -primary alkaline battery – secondary battery - Pb-acid battery.

Green fuel – Hydrogen - production (Photo electrocatalytic and photo catalytic water splitting), construction, working principle and applications in H2 -O2 fuel cells.

Batteries used in E- vehicle: Ni-metal hydride battery, Li-ion Battery - recycling of Li-ion batteries by direct cycling method; environmental effects of different energy storage devices.

(Theory-9)

1. Construction of electrochemical cell.
2. Determination of discharging state of Pb-acid battery by estimating the strength of the acid correlates with specific gravity.
3. Study of performance of a battery using battery analyzing module.

(Laboratory-6)

UNIT III CHEMICAL SENSORS AND CHEMINFORMATICS — 15

Introduction: classification of chemical sensors -principle, construction and working of chemical sensors; pH sensor – Glass electrode; Breath analyzer; Industrial sensor – CO2 sensors- Sensor for health care – Glucose sensor.

Cheminformatics: definition, scope, and significance; applications in the environmental sector – carbon footprint measurements, data analysis and interpretation.

(Theory-9)

1. Determination of the amount of given hydrochloric acid using a pH meter.
2. Calculate the carbon footprint from the provided dataset, analyze the results and draw conclusions.

(Laboratory-6)

UNIT IV	SMART MATERIALS	15

Shape Memory Alloys: introduction - shape memory effect – functional properties of SMAs – types of SMA - Nitinol (Ni-Ti) alloys and its applications.

Chromogenic materials: introduction – types, applications in chemical and biological detection, display technologies, smart windows and light- modulating devices, biomedical and healthcare.

Smart Hydrogels: Introduction - Super Absorbent Polymers (SAP) - preparation, properties and applications of polyacrylic acid and sodium polyacrylate

(Theory-9)

1. Demonstrate the shape memory effect using Nitinol wire.
2. Determination of pH sensitivity of bromothymol blue.
3. Determination of absorption efficiency of hydrogel by using kinetic study.

(Laboratory-6)

UNIT V	NANO CHEMISTRY	15

Introduction – synthesis – top-down process (laser ablation, chemical vapor deposition), bottom-up process (precipitation, electrochemical deposition) - properties of nanomaterials – types – nanotubes -carbon nanotubes, applications of CNT - nanocomposites – general applications of nanomaterials in electronics, information technology, medical and healthcare, energy, environmental remediation, construction and transportation industries.

(Theory-9)

1. Preparation of nano $BaSO_4$ by precipitation method.
2. Demonstrate the efficiency of nano adsorbents in polluted water.

(Laboratory-6)

TOTAL: 75 PERIODS

OUTCOMES:

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

CO1: To examine the role of polymers in different industrial sectors.
CO2: To identify the suitability of batteries for various fields.
CO3: To apply the fundamental principles of chemical sensors, cheminformatics and their applications across various industries.
CO4: To analyze the types of smart materials used in various engineering fields.
CO5: To explore the applications of nanomaterials in various fields, considering their Advantages and limitations.
CO6: To integrate the concepts of chemistry for various engineering applications**.**

TEXT BOOKS:

1. P. C. Jain and Monika Jain, "Engineering Chemistry", Dhanpat Rai Publishing Company Pvt. Ltd., New Delhi, 19th Edition, 2024.
2. Sivasankar B., "Engineering Chemistry", Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2nd reprint, 2012.

REFERENCES:

1. S.S. Dara and S.S. Umare, "A Textbook of Engineering Chemistry,S. Chand &company, New Delhi, 12th Edition, 2022.
2. V.R. Gowarikar, Polymer Science, New Age International Publishers, 4th edition,2021.
3. J. C. Kuriacose and J. Rajaram, "Chemistry in Engineering and Technology", Volume -1 & Volume -2, Tata McGraw-Hill Education Pvt. Ltd., 2010.
4. Barry A. Bunin, Brian Siesel, and J. Bajorath, "Chemoinformatics: Theory, Practice, & Products", Springer, First Edition, 2007.
5. Geoffrey A. Ozin, Andre C. Arsenault and Ludovico Cademartiri, "Nanochemistry: A Chemical Approach to Nanomaterials", RSC publishers, 2nd Edition, 2015.
6. J. Mendham, R. C. Denney, J. D. Barnes, M. J. K. Thomas and B. Sivasankar, "Vogel's Quantitative Chemical Analysis", Pearson Education Pvt. Ltd., 6th edition, 2019.
7. NPTEL course on "Polymers: concepts, properties, uses and sustainability" Prof. Abhijit P Deshpande, IIT-Madras, https://onlinecourses.nptel.ac.in/noc20_ch41/preview
8. NPTEL course on "Electrochemical Energy Storage" Prof. Subhasish Basu Majumder, IIT Kharagpur, https://onlinecourses.nptel.ac.in/noc21_mm34/preview
9. NPTEL course on "Nanotechnology, Science and Applications", Prof. Prathap Haridoss, IIT-Madras, https://onlinecourses.nptel.ac.in/noc22_mm33/preview

LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS

S. No.	Description of Equipment	Quantity
1	Conductivity meter	12 Nos.
2	pH meter	12 Nos.
3	UV-Visible Spectrophotometer	2 Nos.
4	Potentiometer	12 Nos.
5	CAN Enabled BMS unit	4 Nos.
6	Photoreactor	2 Nos.

24EC102	DIGITAL PRINCIPLES AND SYSTEMS DESIGN	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Acquire the knowledge in Digital fundamentals and its simplification methods.
- Familiarize the design of various combinational digital circuits using logic gates.
- Realize various sequential circuits using flip flops.
- Elucidate various semiconductor memories and related technology.
- Build various logic functions using Programmable Logic Devices.

UNIT I	BOOLEAN ALGEBRA AND LOGIC GATES	15

Review of number systems-representation-conversions, Review of Boolean algebra theorems, sum of product and product of sum simplification, canonical forms, min term and max term, Simplification of Boolean expressions-Karnaugh map, Implementation of Boolean expressions using logic gates and universal gates.

List of Exercise/Experiments:

- 1. Implementation of Boolean expression using logic gates.

UNIT II	COMBINATIONAL LOGIC CIRCUITS	15

Design of combinational circuits - Half and Full Adders, Half and Full Subtractors, Binary Parallel Adder – Carry look ahead Adder, Magnitude Comparator, Decoder, Encoder, Priority Encoder, Mux/De-mux, Parity Generator/Checker

List of Exercise/Experiments:

2. Design of adders
3. Design of subtractors.
4. Design of binary adder using IC7483
5. Design of Multiplexers &Demultiplexers.
6. Design of Encoders and Decoders.
7. Implementation of a boolean function using a multiplexer

UNIT III	SEQUENTIAL CIRCUITS	15

Flip flops – SR, JK, T, D, Master/Slave FF – operation and excitation tables, Asynchronous and Synchronous Counters Design - Shift registers, Universal Shift Register.

List of Exercise/Experiments:

8. Design and implementation of 3 bit ripple counters.
9. Design and implementation of 3 bit synchronous counter
10. Design and implementation of shift registers

UNIT IV	SYNCHRONOUS SEQUENTIAL CIRCUITS DESIGN	15

Design of clocked sequential circuits - Moore/Mealy models, state minimization, state assignment, circuit implementation.

UNIT V	MEMORY AND PROGRAMMABLE LOGIC DEVICES	15

Basic memory structure ROM: PROM – EPROM – EEPROM –RAM – Static and dynamic RAM – Programmable Logic Devices: Programmable Logic Array (PLA) – Programmable Array Logic (PAL) – Implementation of combinational logic circuits using PLA, PAL.

TOTAL: 45 PERIODS (THEORY) + 30 PERIODS (LAB) = 75 PERIODS

OUTCOMES:

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

CO1: Apply Boolean algebra to simplify and implement digital circuits.

CO2: Design combinational circuits to meet specific functional requirements using logic gates.

CO3: Demonstrate the operation of counters and shift registers using flip-flops in sequential circuits.
CO4: Analyze synchronous sequential circuits to determine their behavior and performance characteristics.
CO5: Evaluate various types of memory devices, discussing their roles and functionalities in digital systems.
CO6: Construct combinational circuits using Programmable Logic Devices (PLDs) to solve complex digital design problems.

TEXTBOOKS:

1. M. Morris Mano and Michael D. Ciletti, Digital Design, With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6th Edition, Pearson, 2018.
2. S.Salivahanan and S.Arivazhagan,Digital Circuits and Design, 5th Edition, Oxford University Press, 2018.

REFERENCES:

1. A.Anandkumar, Fundamental of digital circuits, 4th Edition, PHI Publication,2016.
2. WilliamKleitz, Digital Electronics-A Practical approach to VHDL, Prentice Hall International Inc, 2012.
3. CharlesH.Roth, Jr. andLarry L. Kinney, Fundamentals of Logic Design, 7th Edition, Thomson Learning, 2014.
4. Thomas L. Floyd, Digital Fundamentals, 11th Edition, Pearson Education Inc, 2017.
5. John.M Yarbrough, Digital Logic: Applications and Design, 1st Edition, Cengage India, 2006.

NPTEL Link:
https://nptel.ac.in/courses/108105132

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS

IC Trainer Kit -15 Nos
ICs each - 30 Nos
7400/ 7404 / 7486 / 7408 / 7432 / 7483 / 7473 / 7411/7474

24MC101	STUDENT INDUCTION PROGRAM (SIP) (Common to all branches of First Year B.E./ B.Tech)	L	T	P	C
		3 Weeks			

OBJECTIVES:

The Course will enable learners to:

- Facilitate the adjustment of new students to the new environment, ensuring they feel comfortable and supported.
- Inculcate the institution's ethos and culture in new students, helping them internalize these values.
- Encourage the building of bonds among students and between students and faculty
- members.
- Expose students to a sense of larger purpose and guide them in their journey of self-exploration.

The various modules or core areas recommended for the 3-week SIP are

Module 1	Universal Human Values I	18
Welcome and Introductions - Aspirations and Concerns - Self- Management - Health - Relationships - Society - Natural Environment - Sum Up - Self-evaluation and Closure.		
Module 2	**Physical Health and Related Activities**	**6**
Special Lecturers: Happy and Healthy lifestyle - Physical Health -Mental Health - Health and Fitness.		
Module 3	**Familiarization of Department/ Branch and Innovation**	**8**
Principal Address - Address by Head of Science and Humanities - Addresses by Respective Department HoDs – Campus Tour – CoE introduction – Introduction of Student Activity Cell (SAC).		
Module 4	**Visit to a Local Area**	**4**
Virtual tour: Government Museum - Theosophical Society - Fort St. George - Ripon Building - Kalakshetra Foundation - Anna Centenary Library - Marina Beach - St. Thomas Mount - Vivekananda House.		
Module 5	**Lectures by Eminent People**	**10**
Special Lecturers: Academics – industry – Careers – Art - Self-management.		
Module 6	**Proficiency Modules**	**30**
Basic Competencies: C Programming, Foundation in Mathematics, Interpersonal Communication.		
Module 7	**Literature / Literary Activities**	**7**
Literary Debate - Creative Writing Workshop - Literature Circle Discussions - Author Study and Presentation.		
Module 8	**Creative Practices**	**10**
Activity: Handicrafts (Painting / Drawing / Pottery / Knitting / Jewellery making, etc.)		
Module 9	**Extra-Curricular Activities**	**10**

Students Activity Cell: Activities from Coding Club – Math Club -- Language Club – Astronomy Club - ECO Club - Photography Club - Tedx Club -Yoga Club.	
Valedictory and Closing Ceremony	**2**
TOTAL: 105 PERIODS	

OUTCOMES:

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

CO1: Achieve a smooth transition where students feel comfortable and confident in their new environment.

CO2: Demonstrate a strong understanding and practice of the institution's ethos and culture within the campus community.

CO3: Build meaningful and supportive relationships with peers and faculty members.

CO4: Develop a clear sense of purpose and engage in self-exploration, leading to a deeper understanding of personal goals and aspirations.

REFERENCES:

https://www.aicte-india.org/sites/default/files/Detailed%20Guide%20on%20Student%20Induction%20program.pdf

24HS111	INTERPERSONAL SKILLS, PSYCHOMETRIC ANALYSIS AND CAREER DEVELOPMENT Employability Enhancement Courses (All Branches)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Evaluate and enhance language proficiency by using SMART Vox to assess communication skills and linguistic abilities.
- Explore future technologies, solve real-time problems, and prepare students for campus recruitment.
- Assess and develop work style, cognitive abilities, emotional intelligence, and work competencies
- Explore placements, internships, higher education options, GATE and CAT exams, and career development.
- Understand career milestones, assess personal skills and personality, and develop effective goal-setting strategies for successful career planning

UNIT I	LANGUAGE PROFICIENCY EVALUATION	3

Identification of Strengths and Weaknesses - Assessing Language Skills (Diagnostic tests and interactive exercises) - Measuring Language Competence (Proficiency Levels) - Identifying Support Needs - Developing Individual Learning Plans - Enhancing Student Experience (Building Confidence) – Evaluation using SMART Vox

UNIT II	CAREER GUIDANCE	3

Future of Engineering- Various aspects of Technology and its Applications - Future of Technologies – Branch Specific emerging technologies - Problems solving through open source - Campus recruitment process.

UNIT III	PSYCHOMETRIC EVALUATION	3

Understanding Behavioural work style (Personality)- Testing of numerical, logical, and verbal reasoning skills (Cognitive Abilities / Aptitude) – Measure of emotional intelligence and interpersonal skills – Determination of Culture Preferences in various workplace scenarios – Evaluation of Work Competencies through targeted games and simulations

UNIT IV	CAREER PREPARATION	

Present Scenario of Engineering - Placement Opportunities - Internship Opportunities - Types of Internships- Higher Education opportunities in India and Abroad - Understanding GATE and CAT Exams - Other Opportunities - Career path development plans.

UNIT V	CAREER VISION AND PLANNING	3

Introduction to career milestones - Overview of the Vision Assessment and its benefits - Psychometric evaluation - Numeracy, Literacy, Visual Reasoning, Algorithmic Thinking - Introduction to a goal-setting model - Identification career pathways aligned with personality profiles - Evaluate personal skills and abilities in various areas

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Assess and improve their English language proficiency using SMART Vox, gaining insights into their communication skills and linguistic competence

CO2: Understand future engineering trends, emerging technologies, importance of solving real-time problems, and the process of campus recruitment.

CO3: Evaluate their behavioral work style, cognitive abilities, emotional intelligence, cultural

preferences, and work competencies.
C**O 4:** Understand the current engineering landscape, placement opportunities, and higher education prospects to develop effective career path plans
CO5: Develop a clear and actionable vision for their future career path.

.

24ME111	IDEA LAB - I (Common to all Branches)	L	T	P	C
		0	0	1	0

The students may be grouped into 3 to 4. The device/Machine/system/component are studied by the students and a final presentation to be done by the students about the study of various devices or machinery at the end of the semester.

OBJECTIVES:

The Course will enable learners to:

- Understand the functionalities and limitation of various machines/equipment
- Demonstrate various operations that can be performed using various Machines

LIST OF EXPERIMENTS

1. Study of fundamental operations of 3D Printer and Scanner with Software.
2. Study of Laser cutting machine.
3. Study of CNC Router machine.
4. Study of Fundamentals of basic circuit design, Soldering and Desoldering.
5. Study of PCB Milling Machine.

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Describe the working of the 3D Printer.0

CO2: Explain the operation of the CNC router and laser cutting machines.

CO3: Explain the basic parts and PCB fabrication process.

CO4: Develop the ability to handle delicate electronic components carefully, minimizing damage during the soldering process.

CO5: Describe the process for converting ideas into prototypes.

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS

S. No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos.
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets

SEMESTER -II

24MA201	LINEAR ALGEBRA AND APPLICATIONS (Common to B.E./B.Tech. CSE, IT, CSD & ADS)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Comprehend the fundamental concepts of matrices.
- Illustrate the basic notions associated with vector spaces and its properties.
- Utilize the Gram-Schmidt ortho normalization process.
- Understand the components and implications for vector spaces by rank-nullity dimension theorem.
- Calculate the eigenvalues and eigenvectors of linear transformations.

UNIT I	MATRICES AND SYSTEM OF LINEAR EQUATIONS	15

Matrices – Row echelon form – Rank of a matrix – System of linear equations – Consistency – Gauss elimination method – Gauss Jordan method.

Experiments using C language:

1. Solve the system of equations using Gauss Elimination method.
2. Solve the system of equations using Gauss Jordan method.

UNIT II	VECTOR SPACES	15

Real and Complex fields – Vector spaces over Real and Complex fields – Subspace – Linear space – Linear independence and dependence (Statement only) – Bases and dimensions.

Experiments using C language:

1. Check whether the given vectors are linearly independent or not.
2. Find the basis and dimension for given vectors.

UNIT III	INNER PRODUCT SPACES	15

Inner product space and norms – Properties – Orthogonal, Orthonormal vectors – Gram-Schmidt ortho normalization process – Least squares approximation.

Experiments using C language:

1. Find the orthogonal vectors using inner product.
2. Find the orthonormal vectors using inner product.

UNIT IV	LINEAR TRANSFORMATION	15

Linear transformation – Range and null space – Rank and nullity – Rank nullity Dimension theorem – Matrix representation of linear transformation – Eigenvalues and eigenvectors of linear transformation.

Experiments using C language:

1. Find the Rank and Nullity of a matrix.
2. Find the eigenvalues and eigenvectors of a matrix.

UNIT V	EIGENVALUE PROBLEMS AND MATRIX DECOMPOSITION	15

Eigenvalue problems – Power method – Jacobi method – Singular value decomposition – QR decomposition.

Experiments using C language:

1. Solve the system of equations using Jacobi method.
2. Find QR decomposition of a matrix.

TOTAL: 75 PERIODS

OUTCOMES:

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

CO1: Solve the system of linear equations using Gauss elimination and Gauss Jordan method.

CO2: Analyze vector spaces to determine their bases and dimensions.
CO3: Apply Gram-Schmidt process to ortho normalize sets of vectors.
CO4: Apply rank nullity theorem to analyse linear transformations.
CO5: Compute the eigenvalues and eigenvectors using singular value decomposition.
CO6: Understand the ideas of least squares approximations and its applications.

TEXTBOOKS:

1. 1. A.H. Friedberg, A. J. Insel, and L. Spence, “Linear Algebra”, Prentice Hall of India, 5th Edition, New Delhi, 2008.
2. Steven J. Leon, “Linear Algebra with Applications”, Pearson Educational International”, 9th Edition, United States of America, 2015.

REFERENCES:

1. 1. G. Strang, “Linear Algebra and its applications”, Thomson (Brooks / Cole), 4th Edition, New Delhi, 2005.
2. 2. C.F. Gerald and P.O. Wheatley, “Applied Numerical Analysis”, 7th Edition, Pearson Education, New Delhi, 2004.
3. Richard Branson, “Matrix Operations”, Schaum's outline series, 1989.
4. Bernard Kolman, R. David R. Hill, “Introductory Linear Algebra”, Pearson Educations, New Delhi, First Reprint, 2009.
5. 5. S. Kumaresan, “Linear Algebra - A geometric approach”, Prentice Hall of India, New Delhi, Reprint, 2010.
6. NPTEL course on "Linear Algebra", by Prof. K. C. Sivakumar, IIT Madras: https://archive.nptel.ac.in/courses/111/106/111106051/##

24CS201	DATA STRUCTURES (Common to CSE, CSD, ADS, CSBS and IT)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- To understand the concepts of List ADT.
- To learn linear data structures – stacks and queues ADTs.
- To understand and apply Tree data structures.
- To understand and apply Graph structures.
- To analyze sorting, searching and hashing algorithms.

UNIT I	LINEAR DATA STRUCTURES – LIST	9+9

Algorithm analysis - running time calculations - Abstract Data Types (ADTs) – List ADT – array- based implementation – linked list implementation – singly linked lists - circularly linked lists - doubly-linked lists – applications of lists – Polynomial Manipulation – All operations (Insertion, Deletion, Merge, Traversal).

List of Exercise/Experiments:

- Array implementation of List ADTs.
- Linked list implementation of List ADTs.

UNIT II	LINEAR DATA STRUCTURES – STACKS, QUEUES	9+9

Stack ADT – Stack Model - Implementations: Array and Linked list - Applications - Balancing symbols - Evaluating arithmetic expressions - Conversion of Infix to postfix expression - Queue ADT – Queue Model - Implementations: Array and Linked list - applications of queues - Priority Queues – Binary Heap – Applications of Priority Queues.

List of Exercise/Experiments:

- Array implementation of Stack and Queue ADTs.
- Linked list implementation of Stack and Queue ADTs.
- Applications of List – Polynomial manipulations
- Applications of Stack – Infix to postfix conversion and expression evaluation.

UNIT III	NON LINEAR DATA STRUCTURES – TREES	9+9

Tree ADT – tree traversals - Binary Tree ADT – expression trees – applications of trees – binary search tree ADT– AVL Tree.

List of Exercise/Experiments:

- Implementation of Binary Trees and operations of Binary Trees.
- Implementation of Binary Search Trees.
- Implementation of Heaps using Priority Queues.

UNIT IV	NON LINEAR DATA STRUCTURES - GRAPHS	9+9

Definition – Representation of Graph – Types of graph - Breadth-first traversal - Depth-first traversal – Topological Sort – Applications of graphs – BiConnectivity – Euler circuits.

List of Exercise/Experiments:

- Graph representation and Traversal algorithms.

UNIT V	SEARCHING, SORTING AND HASHING TECHNIQUES	9+9

Searching- Linear Search - Binary Search - Sorting - Bubble sort - Selection sort - Insertion sort – Hashing - Hash Functions – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing.

List of Exercise/Experiments:

- Implement searching and sorting algorithms.

TOTAL: 45(L) + 45(P) = 90 PERIODS

OUTCOMES:

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

CO1: Analyze algorithms and abstract data types (ADTs).
CO2: Evaluate fundamental data structures.
CO3: Implement linked data structures and its application.
CO4: Apply advanced tree data structures.
CO5: Understand basic graph theory concepts.
CO6: Evaluate various searching and sorting algorithms.

TEXTBOOKS:

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 4th Edition, Pearson Education, 2014.
2. Sartaj Sahni, "Data Structures, Algorithms and Applications in C++", Silicon paper publications, 2004.

REFERENCES:

1. Rajesh K. Shukla, "Data Structures using C and C++", Wiley India Publications, 2009.
2. Narasimha Karumanchi, "Data Structure and Algorithmic Thinking with Python: Data Structure and Algorithmic Puzzles", CareerMonk Publications, 2020.
3. Jean-Paul Tremblay and Paul Sorenson, "An Introduction to Data Structures with Application", McGraw-Hill, 2017.
4. Mark Allen Weiss, "Data Structures and Algorithm Analysis in Java", Third Edition, Pearson Education, 2012.
5. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Second Edition, University Press, 2008.
6. Ellis Horowitz, Sartaj Sahni, Dinesh P Mehta, "Fundamentals of Data Structures in C++", Second Edition, Silicon Press, 2007.
7. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013501578165051392105 84/overview

LIST OF EQUIPMENTS:

1. Systems with Linux/Ubuntu Operating System with gnu C++ compiler

24CS202	JAVA PROGRAMMING (Common to All Branches)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- To explain object oriented programming concepts and fundamentals of Java
- To apply the principles of packages, interfaces and exceptions
- To develop a Java application with I/O streams, threads and generic programming
- To build applications using strings and collections.
- To apply the JDBC concepts

UNIT I	JAVA FUNDAMENTALS	9+9

An Overview of Java - Data Types, Variables, and Arrays – Operators - Control Statements – Class Fundamentals – Declaring objects – Methods – Constructors – this keyword – Overloading methods - Overloading constructors - Access Control – Static – Final

List of Exercise/Experiments:

1. Develop a Java application to generate Electricity bill. You must use one super class called EB Bill and must have two sub classes namely Domestic Bill and Commercial Bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection (i.e domestic or commercial). Compute the bill amount using the following tariff

If the type of the EB connection is domestic, calculate the amount to be paid as follows: First 100 units - Rs. 1 per unit

101-200 units - Rs. 2.50 per unit 201 -500 units - Rs. 4 per unit

> 501 units - Rs. 6 per unit

If the type of the EB connection is commercial, calculate the amount to be paid as follows: First 100 units - Rs. 2 per unit

101-200 units - Rs. 4.50 per unit 201 -500 units - Rs. 6 per unit

> 501 units - Rs. 7 per unit

2. Arrays Manipulations: (Use Methods for implementing these in a Class)

a. Find kth smallest element in an unsorted array

b. Find the sub array with given sum

c. Matrix manipulations – Addition, Subtraction, Multiplication

d. Remove duplicate elements in an Array

e. Accept an integer value N and print the Nth digit in the integer sequence 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 and so on till infinity.

Example: The 11th digit in the sequence 12345678910111213…. is 0.

UNIT II	INHERITANCE, INTERFACES AND EXCEPTION HANDLING	9+9

Inheritance: Inheritance basics, Using super, Method Overriding, Using Abstract Classes, Using final with Inheritance - Package and Interfaces: Packages, Packages and member access, Importing Packages, Interfaces, Static Methods in an Interface – 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.

List of Exercise/Experiments:

1. Develop a Java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages.
2. Develop a Java application with Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor,

Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% of BP for staff club fund. Generate pay slips for the employees with their gross and net salary.
3. Design a Java interface for ADT Stack. Implement this interface using array and built-in classes. Provide necessary exception handling in both the implementations.
4. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains the methods print Area () that prints the area of the given shape and Numberofsides() that prints the number of sides of the given shape.
5. Write a Java program to apply built-in and user defined exceptions.

UNIT III	MULTITHREADING, I/O AND GENERIC PROGRAMMING	9+9

Multithreaded Programming: Creating a Thread, Thread Priorities, Synchronization, Interthread Communication – I/O: I/O Basics, Reading Console Input, Writing Console Output, Reading and Writing Files – Generics: Introduction, Generic class, Bounded Types, Generic Methods, Generic Interfaces, Generic Restrictions.

List of Exercise/Experiments:

1.Write a Java program to read and copy the content of one file to other by handling all file related exceptions.

UNIT IV	STRING HANDLING AND COLLECTIONS	9+9

Lambda Expressions - String Handling – Collections: The Collection Interfaces, The Collection Classes – Iterator – Map - Regular Expression Processing.

List of Exercise/Experiments:

1.String Manipulation:

a. Reversing a set of words and count the frequency of each letter in the string.
b. Pattern Recognition - Find the number of patterns of form 1[0]1 where [0] represents any number of zeroes (minimum requirement is one 0) there should not be any other character except 0 in the [0] sequence in a given binary string.
c. Remove all the occurrences of string S2 in string S1 and print the remaining.
d. Find the longest repeating sequence in a string
e. Print the number of unique string values that can be formed by rearranging the letters in the string S.

2. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
3. Collections:

a. Write a program to perform string operations using ArrayList. Write functions for the following

i. Append - add at end
ii. Insert – add at particular index
iii. Search
iv. List all string starts with given letter

b. Find the frequency of words in a given text.

UNIT V	JDBC CONNECTIVITY	9+9

JDBC – DataSource, Configurations, Connection, Connection Pools, Driver Types, ResultSet, Prepared Statement, Named Parameter, Embedded SQL (Insert, Update, Delete, Join, union etc), ResultSet Navigation, Connection Close and Clean up.

List of Exercise/Experiments:

- Mini Project (using JDBC)

TOTAL: 45(L) + 45(P) = 90 PERIODS

OUTCOMES:

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

CO1: Solve core Java programming concepts.
CO2: Utilize object-oriented programming (OOP) principles.
CO3: Demonstrate competency in handling exceptions and implementing multithreading.
CO4: Develop expertise in input/output (I/O) operations and file handling.
CO5: Apply advanced Java programming concepts with generics and lambda expressions.
CO6: Implement database connectivity using JDBC.

TEXTBOOKS:

1. Herbert Schildt, "Java: The Complete Reference", 11th Edition, McGraw Hill Education, 2019.

REFERENCES:

1. Cay S. Horstmann, Gary Cornell, "Core Java Volume – I Fundamentals", 11th Edition, Prentice Hall, 2019.
2. Paul Deitel, Harvey Deitel, Java SE 8 for programmers, 3rd Edition, Pearson, 2015.
3. Steven Holzner, Java 2 Black book, Dream tech press, 2011.
4. Timothy Budd, Understanding Object-oriented programming with Java, Third Edition, Pearson Education, 2008.
5. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29959473947367270000_shared/overview

LIST OF EQUIPMENTS:

1. Java and Eclipse / NetBeans IDE or Equivalent

24PH201	PHYSICS FOR INFORMATION SCIENCE (Common to B.E CSD, CSE, B.Tech ADS, and IT)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Understand the classical free electron theory and Fermi distribution function
- Relate the theory of laser with its applications in optical fibers
- Solve the Schrodinger's wave equation in one dimensional and three dimensional box
- Gain the basic knowledge in quantum operators and quantum gates
- Comprehend the behavior of semiconductor diodes in various electron devices and nano electronic devices

UNIT I	ELECTRICAL PROPERTIES OF MATERIALS	15

Classical free electron theory - Expression for electrical conductivity and thermal conductivity - thermal conductivity of a bad conductor- Lee's disc method -Effect of temperature on Fermi function - Density of energy states and average energy of an electron at 0 K- Effective mass of electron - Concept of hole.

Semiconductors - Direct and Indirect bandgap semiconductors - Intrinsic Carrier Concentration - Bandgap Determination.

(Theory -9)

1. Determination of Thermal conductivity of a bad conductor - Lee's Disc Method
2. Bandgap determination of intrinsic semiconductor

(Laboratory- 6)

UNIT II	LASER	18

Characteristics of Laser, Spatial and Temporal Coherence - Population inversion – Relation between Einstein's A and B coefficients - Components of Laser - Optical amplification (qualitative) - Semiconductor lasers: Homojunction and Heterojunction- Application: Engineering applications of lasers in data storage (qualitative), Principle of Fiber optics- Fiber optic communication system - Fiber optic sensors (pressure and displacement).

(Theory 9)

1. Determination of divergence of the laser beam
2. Determination of acceptance angle and numerical aperture of an optical fiber
3. Determination of wavelength of semiconductor laser

(Laboratory 9)

UNIT III	QUANTUM THEORY	15

Introduction- Blackbody Radiation - Newton's law of cooling - Planck's quantum theory-matter waves, de-Broglie wavelength, Heisenberg's Uncertainty principle, Time independent and Time- dependent Schrödinger's wave equations, Physical significance of wave function, Particle in a one-dimensional potential box - Particle in a three-dimensional box (qualitative) - degenerate and non-degenerate energy levels- Quantum tunneling - Scanning Tunneling Microscope (STM).

(Theory -9)

1. Determination of emissivity (Newton's law of cooling)
2. Determination of Planck's constant

(Laboratory- 6)

UNIT IV	BASICS OF QUANTUM COMPUTING	12

Quantum Operators: Linear vector spaces - inner product space - Hilbert space - examples Vectors and Tensors: Scalars and vectors, Dirac notations of Bra-Ket notation-Matrix representation of observables and states.

Quantum Computing: Quantum states - classical bits - quantum bits or qubits –

Entanglement and superposition - multiple qubits - Bloch sphere - quantum gates - CNOT gate - Advantages of quantum computing over classical computing.

(Theory -9)

1. Truth table verification of CNOT gate through Virtual Laboratory

(Laboratory-3)

UNIT V	NANOELECTRONIC DEVICES	15

Introduction to Nano materials – synthesis by sol gel method, properties - Quantum confinement - Quantum structures: Density of energy states of quantum wells, quantum wires and quantum dots- band gap of nanomaterials - Quantum dot laser- Single electron phenomena -single electron transistor - Quantum system for information processing.

(Theory -12)

1. Synthesis of Nano-powders by sol-gel method

(Laboratory- 3)

TOTAL: 75 PERIODS

OUTCOMES:

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

CO1: Derive electrical and thermal conductivities using classical free electron theory
CO2: Calculate the electrical conductivity and bandgap in Intrinsic semiconductors
CO3: Associate the basic principles of working of laser and their applications in fiber optics
CO4: Calculate the energy eigen value and eigen function for a particle in a one- dimensional and three dimensional box using Schrodinger wave equations
CO5: Use quantum operators to frame equations for logic gates in Quantum computing
CO6: Relate the quantum properties of nanoscale materials with their applications

TEXTBOOKS:

1. Neil W Ashcroft and N David Mermin, Solid State Physics, Harcourt College Publishers,1976
2. M.N. Avadhanulu and P.G. Kshirsagar, A textbook of Engineering Physics, S. Chand and Company, New Delhi, 2014.
3. David J. Griffiths, Introduction to Quantum Mechanics, 2nd Edition, Pearson Prentice-Hall (2004).
4. Thomas G. Wong, Introduction to Classical and Quantum Computing, Rooted Grove (2022).

REFERENCES:

1. R. A. Serway and J.W. Jewett, Physics for Scientists and Engineers, Ninth Edition, Cengage Learning, 2014.
2. Marikani, Materials Science, PHI Learning Private Limited, Eastern Economy Edition, 2017.
3. R. Wolfson, Essential University Physics, Volume 1 and 2 with Mastering Physics, Global Edition,3rd Edition, Pearson 2017.
4. B.K. Pandey and S. Chaturvedi, Engineering Physics, Cengage Learning India, 2012.
5. Scott Aaronson, Quantum Computing Since Democritus, Cambridge University Press (2014).
6. Mermin, N. David, Making better sense of quantum mechanics. Reports on Progress in Physics 82.1 (2018): 012002.
7. Michael Nielsen, L. Isaac Chuang, Quantum Computation and Quantum Information, Cambridge University Press (2010).
8. NPTEL course on “Introduction to LASER” by Prof. M. R. Shenoy, IIT Delhi : https://onlinecourses.nptel.ac.in/noc24_ph45/preview
9. NPTEL course on “Introduction to Quantum Computing: Quantum Algorithms and Qiskit” by Prof. Prabha Mandayam, Prof. Anupama Ray, Prof. Sheshashayee

24AM201	INTRODUCTION TO ARTIFICIAL INTELLIGENCE (Common to all Branches)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Understand the basics and applications of Artificial Intelligence.
- Apply the basics of Python programming.
- Use python libraries to solve simple problems.
- Understand the different types of Machine Learning algorithms.
- Solve real world problems using AI/ML.
- Explore the various applications in the field of Artificial Intelligence and Machine Learning.

UNIT I	ARTIFICIAL INTELLIGENCE	6+6

Introduction – Types of AI – ANI, AGI, ASI – Narrow, General, Super AI, Examples - AI problems – Production Systems – State space Representation – Applications of AI in various industries.

List of Exercise:

1. Build a simple AI model using python.

UNIT II	BASICS OF PYTHON	6+6

Introduction to Python programming – Arithmetic Operators - values and types - variables, expressions, statements – Functions – Conditionals and Recursion –Iteration.

Lists: Sequence, Mutable, Traversing, Operations, list slices, list methods - Tuples: Immutable, Tuple Assignment, Tuple as Return Values, Comparing and Sorting.

List of Exercises:

1. Compute the GCD of two numbers.
2. Operations on Tuples: a) finding repeated elements, b) slice a tuple c) reverse a tuple d) replace last value of a tuple.

UNIT III	PYTHON LIBRARIES	6+6

Introduction to Numpy - Multidimensional Ndarrays – Indexing – Properties – Constants – Data Visualization: Ndarray Creation – Matplotlib - Introduction to Pandas – Series – Dataframes – Visualizing the Data in Dataframes - Pandas Objects – Data Indexing and Selection – Handling missing data – Hierarchical indexing – Combining datasets – Aggregation and Grouping – Joins- Pivot Tables - String operations – Working with time series – High performance Pandas.

List of Exercises:

1. Download, install and explore the features of R/Python for data analytics
    - Installing Anaconda
    - Basic Operations in Jupyter Notebook
    - Basic Data Handling
2. Working with Numpy arrays - Creation of numpy array using the tuple, Determine the size, shape and dimension of the array, Manipulation with array Attributes, Creation of Sub array, Perform the reshaping of the array along the row vector and column vector, Create two arrays and perform the concatenation among the arrays.
3. Working with Pandas data frames - Series, DataFrame , and Index, Implement the Data Selection Operations, Data indexing operations like: loc, iloc, and ix, operations of handling the missing data like None, Nan, Manipulate on the operation of Null Vaues (is null(), not null(), dropna(), fillna()).
4. Perform the Statistics operation for the data (the sum, product, median, minimum and maximum, quantiles, arg min, arg max etc.).
5. Use any data set compute the mean ,standard deviation, Percentile.

UNIT IV	MACHINE LEARNING	6+6

Introduction – ML Algorithms Overview – Types – Supervised – Unsupervised – Reinforcement Learning – Introduction to Neural Networks – Working of Deep Learning – Applications of DL – Ethical consideration in AI and ML.

List of Exercise:

1. Apply any Machine Learning model to predict the sales in a store.

UNIT V	**CASE STUDIES**	**6+6**

Disease Prediction – Share Price Forecasting – Weather Prediction – Domain Specific Case Studies.

List of Domain Specific Case Studies:

- For CSE & allied: Sentiment analysis of product reviews using machine learning.
- For ECE & allied: Smart homes using AI.
- For EEE: Forecasting of Renewable energy availability during a specified period using AI.
- Civil: Application of ML for crack detection on concrete structures.
- Mech: Predictive Maintenance for CNC Machines Using AI and Machine Learning.

List of Exercise:

1. Build a machine learning model to solve any real-world problem from your domain.

TOTAL: 30(L) + 30(P) = 60 PERIODS

OUTCOMES:

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

CO1: Elaborate the basics and applications of Artificial Intelligence.
CO2: Apply the basics of Python programming to solve problems.
CO3: Use python libraries to solve simple ML problems.
CO4: Outline the different types of Machine Learning algorithms.
CO5: Use Machine Learning Algorithms to solve real world problems.
CO6: Outline the recent developments in the field of Artificial Intelligence.

TEXT BOOKS:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.
2. Jake VanderPlas, "Python Data Science Handbook – Essential tools for working with data", O'Reilly, 2017.
3. Steve Abrams, "Artificial Intelligence and Machine Learning for Beginners: A simple guide to understanding and Applying AI and ML", Independently published, May 14, 2024.

REFERENCES:

1. Vinod Chandra S S, Anand Hareendran S, Artificial Intelligence and Machine Learning, PHI Learning, 2014.
2. Russell, S. and Norvig, P, Artificial Intelligence: A Modern Approach, Third Edition, Prentice Hall, 2010.
3. Ethem Alpaydın, Introduction to Machine Learning, Second Edition, the MIT Press, Cambridge, Massachusetts, London, England.
4. Stephen Marsland, Machine Learning - An Algorithmic Perspective, 2nd Edition, 2015, by Taylor & Francis Group, 2015.
5. Tom M. Mitchell, Machine Learning, McGraw-Hill Science, ISBN: 0070428077
6. Mayuri Mehta, Vasile Palade, Indranath Chatterjee, Explainable AI: Foundations, Methodologies and Applications, Springer, 2023.

LIST OF EQUIPMENTS:

1. Systems with Anaconda, Jupyter Notebook, Python.

24GE201	TAMILS AND TECHNOLOGY (Common to all branches of First Year B.E. / B. Tech)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Recognize the historical significance of weaving and pottery technologies in ancient Tamil civilization.
- Highlight the concepts of design and construction technology during the Sangam age.
- Provide an overview of manufacturing technology and its role in Tamil society.
- Illustrate the agricultural and irrigation techniques employed in ancient Tamil society.
- Promote scientific Tamil and Tamil computing.

UNIT I	WEAVING AND CERAMIC TECHNOLOGY	3

Weaving Industry during Sangam Age – Ceramic technology – Black and Red Ware Potteries (BRW) – Graffiti on Potteries.

UNIT II	DESIGN AND CONSTRUCTION TECHNOLOGY	3

Designing and Structural construction House & Designs in household materials during Sangam Age - Building materials and Hero stones of Sangam age – Details of Stage Constructions in Silappathikaram - Sculptures and Temples of Mamallapuram – Great Temples of Cholas and other worship places - Temples of Nayaka Period - Type study (Madurai Meenakshi Temple)- Thirumalai Nayakar Mahal - Chetti Nadu Houses, Indo - Saracenic architecture at Madras during British Period.

UNIT III	MANUFACTURING TECHNOLOGY	3

Art of Ship Building - Metallurgical studies - Iron industry - Iron smelting,steel -Copper and gold-Coins as source of history - Minting of Coins – Beads making-industries Stone beads - Glass beads - Terracotta beads -Shell beads/ bone beats - Archeological evidences - Gem stone types described in Silappathikaram.

UNIT IV	AGRICULTURE AND IRRIGATION TECHNOLOGY	3

Dam, Tank, ponds, Sluice, Significance of Kumizhi Thoompu of Chola Period, Animal Husbandry - Wells designed for cattle use - Agriculture and Agro Processing - Knowledge of Sea – Fisheries – Pearl - Conche diving - Ancient Knowledge of Ocean - Knowledge Specific Society.

UNIT V	SCIENTIFIC TAMIL & TAMIL COMPUTING	3

Development of Scientific Tamil - Tamil computing – Digitalization of Tamil Books – Development of Tamil Software – Tamil Virtual Academy – Tamil Digital Library – Online Tamil Dictionaries – Sorkuvai Project.

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Identify the role of weaving and ceramic technology in ancient Tamil Culture.

CO2: Assess the design and construction technology ideas in the current Tamil society.

CO3: Identify the different types of manufacturing technology used in Tamil society and their significance.

CO4: Classify agricultural and irrigation technologies in ancient Tamil society and its current relevance.

CO5: Discuss the fundamentals of scientific Tamil and Tamil computing.

REFERENCES:

1. தமிழக வரலாறு – மக்களும் பண்பாடும் – கே.கே. பிள்ளை (வெளியீடு: தமிழ்நாடு பாடநூல் மற்றும் கல்வியியல் பணிகள் கழகம்).
2. கணினித் தமிழ் – முனைவர் இல. சுந்தரம். (விகடன் பிரசுரம்).

3. கீழடி – வைகை நதிக்கரையில் சங்ககால நகர நாகரிகம் (தொல்லியல் துறை வெளியீடு)
4. பொருநை – ஆற்றங்கரை நாகரிகம். (தொல்லியல் துறை வெளியீடு)

5. Social Life of Tamils (Dr.K.K. Pillay) A joint publication of TNTB & ESC and RMRL – (in print)
6. Social Life of the Tamils - The Classical Period (Dr.S .Singaravelu) (Published by: International Institute of Tamil Studies.
7. Historical Heritage of the Tamils (Dr.S.V.Subatamanian, Dr.K.D.Thirunavukkarasu) (Published by: International Institute of Tamil Studies).
8. The Contributions of the Tamils to Indian Culture (Dr. M. Valarmathi) (Published by:International Institute of Tamil Studies.)
9. Keeladi - 'Sangam City Civilization on the banks of river Vaigai' (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
10. Studies in the History of India with Special Reference to Tamil Nadu (Dr. K. K. Pillay) (Published by: The Author)
11. Porunai Civilization (Jointly Published by: Department of Archaeology & Tamil Nadu Text Book and Educational Services Corporation, Tamil Nadu)
12. Journey of Civilization Indus to Vaigai (R. Balakrishnan) (Published by: RMRL).

24ME211	IDEA LAB – II (Common to all Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Develop hands-on experience and practical application of theoretical knowledge.
- Develop their ability to explain the process involved.

LIST OF EXPERIMENTS

1. Printing of a 3D part.
2. Scanning of a 3D part.
3. Design and fabrication of press fit object using laser cutting machine.
4. Design and fabrication of 3D part using CNC Router.
5. Design and fabrication of simple PCB.
6. Soldering and desoldering of given electronic circuit.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Analyze the latest manufacturing methods in advancements and technologies related to their field.

CO2: Understand the operations of a laser cutting machine and CNC Router.

CO3: Analyze the process of design and fabrication of PCB and Soldering operations

CO4: Develop technical proficiency and problem-solving abilities, making more competent and confident in their field.

CO5: Develop themselves with the skills needed to address industry-specific Problems

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS

S.No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos.
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets

24CS211	INNOVATION AND CREATIVITY SKILLS AND DEVELOPMENT Employability Enhancement Courses (All Branches)	L	T	P	C
		1	0	0	1

OBJECTIVES:

The Course will enable learners to:

- Understand study plans, co-curricular activities, programming skills, recruitment test patterns, and hiring strategies through national qualifiers and hackathons.
- Equip students with strategies for higher education, resume enhancement, project management, and securing internships
- Understand entrepreneurship fundamentals, including key differences, global hubs, business ideas, and scalability.
- Develop essential entrepreneurial skills such as opportunity recognition, patience, risk management, communication, persistence, and leadership
- Understand life, success, self-confidence, health, scientific heritage, personal counseling, and cybercrime awareness

UNIT I	STEPPING STONE – ENGINEERING CAREERS AND SKILL DEVELOPMENT	3

Study Plans and Resources - Identification of key resources and job opportunities - career prospects and academic growth through co-curricular activities - importance of programming/coding skills - Overview of test patterns and essential skills for popular campus recruiters - Comparison of IT Services, Dream, and Super Dream offers and their recruitment processes - National Qualifier Tests and their impact on hiring processes – Overview of Corporate contests and hackathons (e.g., TCS Codevita, HackerRank)

UNIT II	STEPPING STONE – HIGHER EDUCATION AND CAREER DEVELOPMENT	3

Overview of higher education opportunities: GATE, GRE, GMAT, XAT, CAT, MAT - Exam formats, preparation strategies, and timelines - Resume Enhancement Strategies - Project Management - Steps to develop projects from proposal to prototype - Internship Pathways - Strategies for maximizing internship experiences for career advancement

UNIT III	FUNDAMENTALS OF ENTREPRENEURSHIP: FROM IDEAS TO VENTURES	3

Introduction to Entrepreneurship – Intrapreneur vs. entrepreneur - Roles and Contributions - Global Entrepreneurship Hubs - Overview of Key Global Locations - Idea vs. Commercial Value - Transforming Ideas into Viable Business Models - Characteristics of Successful Business Ideas - Understanding Market Competition - Basics of Copyrights and Intellectual Property - Scalability in Business Ventures - Strategies for Scaling a Business

UNIT IV	HUMAN SKILLS FOR ENTREPRENEURSHIP	3

Identifying and capitalizing on business opportunities - Case studies and anecdotes - Patience and Risk Management - The role of patience in entrepreneurial success and decision-making - Effective Communication - Techniques for clear and persuasive communication - Importance of communication in building and leading teams - Leadership qualities and their impact on entrepreneurial ventures - Analyzing success and failure stories

UNIT V	FOUNDATIONS OF PERSONAL DEVELOPMENT AND WELL-BEING	3

Understanding Life and Success - Self-Confidence and Fear - Practical strategies for enhancing self-esteem - Adolescent Issues - Health Management - Basics of a balanced diet -

Benefits of physical activity - Scientific Heritage of India - Overview of India's scientific achievements and contributions - Cyber Crime Awareness - Types and prevention strategies.	
	TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Create study plans, value co-curricular activities, develop programming skills, and navigate for career advancement

CO2: Understand about higher education options, resume enhancement, project management, and securing internships

CO3: Learn entrepreneurship skills and strategies to develop successful business ideas.

CO4: Develop key entrepreneurial skills like opportunity recognition, risk management, and leadership through real-world examples

CO5: Explore personal development, health management, scientific heritage, and cybercrime awareness.

24MC201	Yoga for Stress Management (Common to all branches of First Year B.E. / B. Tech)	L	T	P	C
		0	0	1	0

OBJECTIVES:

The Course will enable learners to:

- Understanding the different types of stress and managing stress.
- Develop an understanding of practicing yoga
- Learning to do asanas, including sitting, standing and lying postures

UNIT I	Stress Management	3

Definition of Stress - Stress in Daily Life - Impact of Stress on Life - Identifying the Causes of Stress - Symptoms of Stress - Managing Stress (Habits, Tools, Training, Professional Help) - Complications of Stress Mismanagement - The Importance of Sleep for Mental Wellness - Connection Between Sleep and Digestion.

UNIT II	Introduction to Yoga	3

Meaning and Definition of Yoga - Aims and Objectives of Yoga - Guidelines for Practicing Asanas - Benefits of Yoga

UNIT III	Different Asanas	3

Methods of Performing Asanas - Pranayama - Suryanamaskar Asanas - Sitting Postures: Uttanpadasana, Paschimottanasana ,Janu Sirsasana , Baddha Konasana - Shishupal Asana – Vajrasana

UNIT IV	Standing Postures	3

Uttanasana -Trikonasana -Vrikshasana -Tadasana - Superbrain asana

UNIT V	Lying Postures	3

Pavana Muktasana - Pada Sanchalanasana – Jhulana Lurhakanasana -Dhanurasana – Marjaryasana. BitilasanaDictionaries – Sorkuvai Project.

TOTAL: 15 PERIODS

OUTCOMES:

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

CO1: Relieve stress and achieve mental wellness.

CO2: Experience the benefits of yoga

CO3: Keep self and body healthy

REFERENCES:

1. Iyengar, Bellur Krishnamukar Sundara. "Light on yoga." (1965).
2. Desikachar, Tirumalai Krishnamacharya Venkata. The heart of yoga: Developing a personal practice. Simon and Schuster, 1999.
3. Davis, Martha, Elizabeth Robbins Eshelman, and Matthew McKay. The relaxation and stress reduction workbook. New Harbinger Publications, 2008.
4. Krishnamacharya, Tirumalai, et al. "Yoga makaranda: The nectar of yoga." Swathi Soft (2013).

24MC105	ENVIRONMENTAL SCIENCE AND SUSTAINABILITY (Common to all Branches)	L	T	P	C
		2	0	0	MC

OBJECTIVES:

The Course will enable learners to:

- To gain knowledge of the environment and various natural resources.
- To identify the Scientific and Technological solutions to pollution issues and Waste management.
- To understand the significance of the conservation of biodiversity.
- To recognize the needs and benefits of sustainability and its management.
- To comprehend the effects of human population on the environment.

UNIT I	NATURAL RESOURCES	7

Definition, scope and importance of environment – need for public awareness. Introduction to natural resources - types - forest resources: use and over-exploitation, deforestation and its impacts, food resources: effects of modern agriculture, organic farming, renewable energy sources - solar, wind, geothermal, tidal, OTE and biomass. field activity -tree plantation

UNIT II	POLLUTION AND WASTE MANAGEMENT	7

Pollution - definition –causes, effects and control measures of (a) air pollution (b) water pollution(c) soil pollution (d) noise pollution (e) nuclear hazards - nuclear accidents and holocaust - role of an individual in prevention of pollution –case studies.

Waste management- municipal solid wastes, E- waste, plastic waste.

Field study – Solid waste management of the institution

UNIT III	BIODIVERSITY AND ITS CONSERVATION	6

Biodiversity: types – values of biodiversity, India as a mega-diversity nation – hot-spots of biodiversity – threats to biodiversity – endangered and endemic species, extinct, rare, vulnerable species of India – conservation of biodiversity: in-situ and ex-situ method.

Field study – Biodiversity of the institution

UNIT IV	SUSTAINABILITY AND MANAGEMENT	5

Sustainability-concept, needs and challenges- circular economy - sustainable development goals-concept of carbon footprint, environmental impact assessment, clean development mechanism, solutions.

Field study – Alternate energy sources and its impacts

UNIT V	HUMAN POPULATION	5

Introduction - population growth, variation among nations, population explosion, environment and human health – endemic/epidemic/pandemic – role of information technology in environment and human health.

Case Study – Pandemics of 21st century

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: To investigate and use conservational practices to protect natural resources.

CO2: To identify the causes of pollutants and illustrate suitable methods for pollution abatement.

CO3: To analyze the values of biodiversity and its conservational methods.

CO4: To classify suitable sustainable development practices and apply it in day-to-day life.

CO5: To assess the impacts of human population and suggest suitable solutions.
CO6: To develop innovative solutions and strategies to address sustainability challenges.

TEXTBOOKS:

1. Anubha Kaushik and C.P. Kaushik, "Perspectives in environmental studies", New Age International Publishers, 8th edition, 2024.
2. Benny Joseph, Environmental Science and Engineering, McGraw-Hill, 1st edition, 2017.
3. Gilbert M. Masters, Introduction to Environmental Engineering and Science, Pearson Education, 3rd edition, 2014.
4. Erach Bharuch, Textbook of Environmental Studies for Undergraduate Courses, Universities Press(I) Pvt. Ltd., 3rd edition, 2021.

REFERENCES:

1. 1. William P. Cunningham and Mary Ann Cunningham Environmental Science: A Global Concern, McGraw Hill, 14th edition, 2017.
2. Rajagopalan, R, Environmental Studies-From Crisis to Cure, Oxford University Press, 2015.
3. G. Tyler Miller and Scott E. Spoolman, ―Environmental Science, Cengage Learning India Pvt, Ltd., Delhi, 2014.
4. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall, 2012.
5. Bradley. A.S; Adebayo, A.O. and Maria, P. Engineering applications in sustainable design and development, Cengage learning, 2015.
6. 6. Environment Impact Assessment Guidelines, Notification of Government of India, 2006 and subsequent amendments, 2022

SEMESTER – III

24MA301	DISCRETE MATHEMATICS	L	T	P	C
		3	1	0	4

OBJECTIVES:

The Course will enable the learners to:

- Describe the arguments using connectives and rules of inference.
- Introduce the basic concept of counting and generating functions.
- Construct recurrence relations for mathematical models.
- Define the graphs and its models.
- Understand the concept of group theory, Lattices and Boolean algebra.

UNIT I	LOGIC AND PROOFS	12

Propositional logic - Propositional equivalences - Predicates and quantifiers - Nested quantifiers - Rules of inference - Introduction to proofs - Proof methods and strategy.

UNIT II	COMBINATORICS	12

Mathematical induction - Strong induction and well ordering - The basics of counting - The pigeonhole principle - Permutations and combinations - Recurrence relations - Solving linear recurrence relations - Generating functions - Inclusion and exclusion principle and its applications.

UNIT III	GRAPHS	12

Graphs and graph models - Graph terminology and special types of graphs - Matrix representation of graphs and graph isomorphism - Connectivity - Euler and Hamilton paths.

UNIT IV	ALGEBRAIC STRUCTURES	12

Algebraic systems - Semi groups and monoids - Groups - Subgroups - Homomorphism‘s - Normal subgroup and cosets - Lagrange‘s theorem - Definitions and examples of Rings and Fields.

UNIT V	LATTICES AND BOOLEAN ALGEBRA	12

Partial ordering -Posets- Lattices as posets- Properties of lattices - Lattices as algebraic systems - Sub lattices - Direct product and homomorphism - Some special lattices - Boolean algebra.

TOTAL: 60 PERIODS

OUTCOMES:

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

CO1: Examine the validity of the arguments.

CO2: Apply various proof techniques and principles using analytic and combinatorial methods.

CO3: Develop the recurrence relation for the sequence.

CO4: Implement graph theory techniques to solve real time problems

CO5: Apply the concepts of groups, rings, and fields in solving algebraic problems.

CO6: Solve problems in Lattices and Boolean algebra.

TEXT BOOK:

1.K. H. Rosen, "Discrete Mathematics and its Applications", 8th Edition, Tata McGraw Hill Pub. Co. Ltd., New Delhi, Special Indian Edition, 2021.

2.J. P. Tremblay, and R. Manohar. " Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2017.

REFERENCES:

1.R.P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", 5th Edition, Pearson Education Asia, New Delhi, Reprint 2019.

2.S. Lipschutz, and Mark Lipson., "Discrete Mathematics", Schaum's Outlines, Tata McGraw Hill Pub. Co. Ltd., New Delhi, 4th Edition, 2021.

3.T. Koshy, "Discrete Mathematics with Applications", Elsevier Publications, 1st Edition, 2014.

24GE301	UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY	L	T	P	C
		2	1	0	3

OBJECTIVES:

The Course is designed to:

- Development of a holistic perspective based on self-exploration about themselves (human beings), family, society and nature/existence.
- Understanding (or developing clarity) of the harmony in the human being, family, society, and nature/existence
- Strengthening of self-reflection.
- Development of commitment and courage to act

UNIT I	NEED, BASIC GUIDELINES, CONTENT AND PROCESS FOR VALUE EDUCATION	9

Purpose and motivation for the course - recapitulation from Universal Human Values-I - SelfExploration–what is it? – It's content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration - Continuous Happiness and Prosperity- A Look at Basic Human Aspirations - Right understanding, Relationship and Physical Facility- the basic requirements for the fulfilment of aspirations of every human being with their correct priority - Understanding Happiness and Prosperity Correctly- A critical appraisal of the current scenario - Method to fulfil the above human aspirations: understanding and living in harmony at various levels. Practice sessions: To discuss natural acceptance in human beings as the innate acceptance of living with responsibility (living in relationship, harmony, and co-existence) rather than as arbitrariness in choice based on liking-disliking.

UNIT II	UNDERSTANDING HARMONY IN THE HUMAN BEING – HARMONY IN MYSELF!	9

Understanding human beings as a co-existence of the sentient 'I' and the material 'Body' - Understanding the needs of Self ('I') and 'Body' - happiness and physical facility -Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer) - Understanding the characteristics and activities of 'I' and harmony in 'I' - 'Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, the meaning of Prosperity in detail - Programs to ensure Sanyam and Health. Practice sessions: To discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss programs for ensuring health vs dealing with the disease.

UNIT III	UNDERSTANDING HARMONY IN THE FAMILY AND SOCIETY HARMONY IN HUMAN-HUMAN RELATIONSHIP	9

Understanding values in a human-human relationship - the meaning of Justice (nine universal values in relationships) and the program for its fulfillment to ensure mutual happiness -Trust and Respect as the foundational values of relationship - Understanding the meaning of Trust; Difference between intention and competence - Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in a relationship - Understanding the harmony in the society (society being an extension of the family) - Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals - Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family. Practice sessions: To reflect on relationships in family, hostel and institute as extended family, real-life examples, teacher-student relationship, the goal of education etc. Gratitude as a universal value in relationships. Discuss scenarios. Elicit examples from students' lives.

UNIT IV	UNDERSTANDING HARMONY IN NATURE AND EXISTENCE - WHOLE EXISTENCE AS COEXISTENCE	9

Understanding the harmony in Nature - Interconnectedness and mutual fulfillment among the four orders of nature- recyclability and self-regulation in nature - Understanding Existence as the

Coexistence of mutually interacting units in all-pervasive Space - Holistic perception of harmony at all levels of existence. Practice sessions: To discuss human beings as the cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc

UNIT V	IMPLICATIONS OF THE ABOVE HOLISTIC UNDERSTANDING OF HARMONY IN PROFESSIONAL ETHICS	9

Natural acceptance of human values - Definitiveness of Ethical Human Conduct - Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order - Competence in professional ethics: a. Ability to utilize professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people-friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for the above production systems - Case studies of typical holistic technologies, management models and production systems - Strategy for the transition from the present state to Universal Human Order: a. At the level of the individual: as socially and ecologically responsible engineers, technologists, and managers b. At the level of society: as mutually enriching institutions and organizations - Sum up. Practice Exercises and Case Studies e.g. To discuss the conduct as an engineer or scientist etc.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop self-awareness and a deeper understanding of their surroundings, including family, society, and nature

CO2: Identify and resolve inner conflicts based on natural acceptance.

CO3: Become more responsible towards life, and handle problems with sustainable solutions by considering human relationships and natural harmony.

CO4: Enhance their critical thinking and analyzing skills.

CO5: Develop a stronger commitment towards human values, relationships, and societal well-being.

CO6: Apply what they have learnt in different day-to-day settings in real life, and take the initial steps towards integrating these values into daily life.

TEXT BOOK:

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, First edition: New Delhi, 2010, ISBN: 978-81-7446-781-2 Reprint: 2010, 2011, 2013, 2016, 2018, 2019 (Second edition: New Delhi, 2019, ISBN: 978-93-87034-47-1)
2. A foundation course in Human Values and Professional Ethics RR Gaur, R Asthana, GP Bagaria, Third revised edition: New Delhi, 2023, ISBN: 978-81-957703-7-3 (Printed Copy), ISBN: 978-81-957703-6-6 (e-book), UHV Publications

REFERENCES:

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful-E. F Schumacher.
6. Slow is Beautiful-Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India – by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi

24CS301	COMPUTER ORGANIZATION AND ARCHITECTURE (Common to CSE, IT,CSBS and AIML)	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Describe the basic principles and operations of digital computers.
- Design arithmetic and logic unit for various fixed and floating point operations
- Construct pipeline architectures for RISC processors.
- Explain various memory systems & I/O interfacings
- Discuss parallel processor and multi-processor architectures

UNIT I	COMPUTER FUNDAMENTALS	9

Computer Types - Functional Units — Basic Operational Concepts - Number Representation and Arithmetic Operations - Performance Measurement - Instruction Set Architecture - Memory Locations and Addresses - Instructions and Instruction Sequencing - Addressing Modes.

UNIT II	COMPUTER ARITHMETIC	9

Addition and Subtraction of Signed Numbers - Design of Fast Adders - Multiplication of Unsigned Numbers - Multiplication of Signed Numbers - Fast Multiplication - Integer Division - Floating-Point Numbers – Representation and Operations.

UNIT III	BASIC PROCESSING UNIT AND PIPELINING	9

Basic Processing Unit: Concepts - Instruction Execution - Hardware Components - Instruction Fetch and Execution Steps -Control Signals - Hardwired Control.

Pipelining: Basic Concept - Pipeline Organization- Pipelining Issues - Data Dependencies - Memory Delays - Branch Delays - Resource Limitations - Performance Evaluation -Superscalar Operation.

UNIT IV	I/O AND MEMORY	9

Input/Output Organization: Bus Structure - Bus Operation - Arbitration - The Memory System: Basic Concepts - Semiconductor RAM Memories - Read-only Memories - Direct Memory Access - Memory Hierarchy - Cache Memories - Performance Considerations - Virtual Memory - Memory Management Requirements - Secondary Storage.

UNIT V	PARALLEL PROCESSING AND MULTICORE COMPUTERS	9

Parallel Processing: Use of Multiple Processors - Symmetric Multiprocessors - Multithreading and Chip Multiprocessors - Clusters - Nonuniform Memory Access Computers - Vector Computation - Multicore Organization.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the basic principles and operations of digital computers.

CO2: Analyse the performance of computers by identifying factors that contribute to performance.

CO3: Compare various I/O methods and understand memory management principles.

CO4: Explain data flow in arithmetic algorithms.

CO5: Demonstrating the concept of parallelism in hardware and software.

CO6: Design hardware to solve computationally intensive problems.

TEXT BOOKS:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization, Tata McGraw Hill, Sixth edition, 2012.
2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface, 5th edition, Morgan Kaufmann, 2013.

REFERENCES:

1. John P.Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012.
2. David A. Patterson and John L. Hennessy Computer Organization and Design-The Hardware/Software Interface, 6th edition, Morgan Kaufmann, 2021.
3. John L. Hennessy and David A. Patterson, Computer Architecture – A Quantitate Approach, Morgan Kaufmann / Elsevier Publishers, Fifth Edition,2012.

24CS302	ADVANCED JAVA PROGRAMMING (Lab Integrated) (Common to All Branches)	L	T	P	C
		3	0	3	4.5

OBJECTIVES:

The Course will enable learners to:

- Gain a comprehensive understanding of the Java Collections Framework and its various interfaces and implementations.
- Learn the details of Java I/O streams and utility classes for managing dates, numbers, and currencies.
- Develop a thorough understanding of the Stream API introduced in Java 8 and its various operations.
- Explore advanced object serialization and string tokenizing techniques, including pattern matching with regular expressions.
- Understand advanced Stream API features and gain proficiency in using regular expressions for text processing.

UNIT I	COLLECTIONS FRAMEWORK AND UTILITY CLASSES	9+9

Introduction to Collections Framework - Collection Interface- Methods in Collection Interface - Iterable and Iterator Interfaces - List Interface- ArrayList - LinkedList - Set Interface - HashSet- LinkedHashSet - TreeSet - Map Interface - HashMap -LinkedHashMap- TreeMap - Queue Interface -PriorityQueue - Deque Interface - Utility Classes.

List of Experiments

1. Write a program that measures the time taken for insertion, deletion, and search operations on ArrayList, LinkedList, HashSet, and TreeSet for varying sizes of input data.
2. Implement a custom data structure that combines features of a list and a set.
3. Write a Java program to create a HashMap where the keys are strings, and the values are integers Add five key-value pairs to the map. Print all the keys and values in the map. Remove an entry by key. Update the value associated with a specific key. Check if the map contains a specific key and a specific value.

UNIT II	DATE HANDLING AND SERIALIZATION	9+9

Date – Calendar – Comparable interface – Observer Interface — Serialization – Dates - Numbers, and Currency - Working with Dates - Numbers and Currencies - Object Serialization - Serializable Interface - Writing and Reading Serializable Objects -Transient Keyword- SerialVersionUID.

List of Experiments

1. Create a class representing a complex object with nested data structures. Serialize the object to a file, then deserialize it back and verify that the object remains intact.
2. Write a program that formats dates and currencies according to different locales.
3. Create a class hierarchy representing different types of objects (e.g., Person, Employee). Serialize instances of these classes to a file using object serialization.

UNIT III	STREAM API AND FUNCTIONAL PROGRAMMING PARADIGMS	9+9

Overview of Stream API - Importance of Stream API in Java 8 and Beyond – Functional Programming Concepts - Creating Streams - Stream Interface Methods - Stream Operations - Intermediate Filtering (filter)-Mapping (map, flatMap)-Sorting (sorted)-Distinct (distinct) - Limit and Skip (limit, skip) - Terminal Operations -Collecting Results (collect) - Reducing and Summarizing (reduce, summaryStatistics)-Iterating (forEach) - Matching and Finding (anyMatch, allMatch, noneMatch, findFirst, findAny) -Counting (count).

List of Experiments

1. Write a program that performs stream operations like filtering, mapping, and reducing.
2. Create an infinite stream generator that generates prime numbers. Implement methods to check for primality and generate the next prime number.
3. Write a program that reads a text file containing sentences. Tokenize each sentence into words, filter out stopwords, and print the remaining words.

UNIT IV	ADVANCED STRING PROCESSING AND I/O TECHNIQUES	9+9

String Tokenizer – Parsing - Tokenizing and Formatting - Locating Data via Pattern Matching,

Tokenizing - Streams - Types of Streams - The Byte-stream I/O hierarchy - Character Stream Hierarchy – Random Access File class – the java.io. Console Class - Advanced I/O - Piped Streams (PipedInputStream and PipedOutputStream) – SequenceInputStream -PushbackInputStream and PushbackReader.

List of Experiments

1. Write a program that reads a text file and tokenizes it into sentences using the StringTokenizer class.
2. Implement a java program that allows users to open a text file, navigate through it using random access, insert, delete, and modify text at specific positions within the file.
3. Implement a program that uses advanced I/O techniques like PipedInputStream, PipedOutputStream, SequenceInputStream, and PushbackInputStream.

UNIT V	ADVANCED STREAM FEATURES AND REGULAR EXPRESSIONS	9+9

Importance and Use Cases of Advanced Stream Features - Creating Custom Streams -Stream Generators (Stream.generate, Stream.iterate) - Infinite Streams - Using Spliterators – Advanced Stream Operations - FlatMapping - Chaining Stream Operations - Stream Peeking (peek) - Advanced Filtering Techniques - Introduction to Regular Expressions - Character Classes - Quantifiers - Pattern Matching - Groups and Capturing - Regex in Java - java.util.regex Package Pattern Class - Matcher Class - String Manipulation with Regex - Splitting Strings - Replacing Text (replaceAll, replaceFirst) - Replacing with Backreferences.

List of Experiments

1. Implement custom stream generators using Stream.generate and Stream.iterate methods.
2. Write a program that demonstrates advanced stream operations like flatMapping, chaining stream operations, and peeking.
3. Develop a program that utilizes regular expressions to perform string manipulation tasks such as splitting strings, replacing text, and extracting specific patterns.

TOTAL: 45+45 = 90 PERIODS

OUTCOMES:

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

CO1: Implement various data structures by utilizing core Java features and libraries

CO2: Demonstrate proficiency in handling Java I/O operations, including file manipulation for efficient data storage and retrieval.

CO3: Apply and Analyze the Stream API for functional programming and data processing.

CO4: Implement advanced object serialization for complex data structures.

CO5: Utilize regular expressions for text parsing and string manipulation.

CO6: Build applications using advanced Java programming techniques.

TEXT BOOK:

3. Cay S. Horstmann, "Core Java Volume I--Fundamentals," 12th Edition, 2019.
4. Joshua Bloch, "Effective Java," 3rd Edition, 2018.
5. Raoul-Gabriel Urma, "Java 8 in Action: Lambdas, Streams, and Functional-Style Programming," 1st Edition, 2014.
6. Herbert Schildt, "Java: The Complete Reference," 11th Edition, 2018.
7. Alan Mycroft and Martin Odersky, "Programming in Scala," 4th Edition, 2020.

REFERENCES:

11. Bruce Eckel, "Thinking in Java," 4th Edition, 2006.
12. Herbert Schildt, "Java: A Beginner's Guide," 8th Edition, 2019.
13. Richard Warburton, "Java 8 Lambdas: Pragmatic Functional Programming," 1st Edition, 2014.

LIST OF EQUIPMENTS:

JDK/Eclipse

24CS303	DATABASE MANAGEMENT SYSTEMS (Lab Integrated)	L	T	P	C
		3	0	3	4.5

COURSE OBJECTIVES:

The Course will enable the learners to:

- Understand the basic concepts of Data Modeling and Database Systems.
- Understand SQL and effective relational database design concepts.
- Learn relational algebra, calculus and normalization.
- Know the fundamental concepts of transaction processing, concurrency control techniques, recovery procedure and data storage techniques.
- Understand query processing, efficient data querying and advanced databases.

UNIT I	DATABASE CONCEPTS	9+9

Concept of Database and Overview of DBMS - Characteristics of databases -Data Models, Schemas and Instances - Three-Schema Architecture - Database Languages and Interfaces- Introductions to data models types- ER Model- ER Diagrams - Enhanced ER Model - reducing ER to table Applications: ER model of University Database Application – Relational Database Design by ER- and EER-to-Relational Mapping.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list

a) Inventory Management for a EMart Grocery Shop
b) Society Financial Management
c) Cop Friendly App – Eseva
d) Property Management – eMall
e) Star Small and Medium Banking and Finance

- Build Entity Model diagram. The diagram should align with the business and functional goals stated in the application.

UNIT II	STRUCTURED QUERY LANGUAGE	9+9

SQL Data Definition and Data Types – Constraints – Queries – INSERT, UPDATE, and DELETE in SQL - Views - Integrity Procedures, Functions, Cursor and Triggers - Embedded SQL - Dynamic SQL.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list and do the following exercises.

a) Inventory Management for a EMart Grocery Shop
b) Society Financial Management
c) Cop Friendly App – Eseva
d) Property Management – eMall
e) Star Small and Medium Banking and Finance

1. Data Definition Commands, Data Manipulation Commands for inserting, deleting, updating and retrieving Tables and Transaction Control statements
2. Database Querying – Simple queries, Nested queries, Sub queries and Joins
3. Views, Sequences, Synonyms
4. Database Programming: Implicit and Explicit Cursors
5. Procedures and Functions
6. Triggers
7. Exception Handling

UNIT III	RELATIONAL ALGEBRA, CALCULUS AND NORMALIZATION	9+9

Relational Algebra – Operations - Domain Relational Calculus- Tuple Relational Calculus - Fundamental operations.
Relational Database Design - Functional Dependency – Normalization (1NF, 2NF 3NF and BCNF) – Multivalued Dependency and 4NF –Joint Dependencies and 5NF - De-normalization.

List of Exercise/Experiments

1. Case Study using real life database applications anyone from the following list
   a) Inventory Management for a EMart Grocery Shop
   b) Society Financial Management
   c) Cop Friendly App – Eseva
   d) Property Management – eMall
   e) Star Small and Medium Banking and Finance.

Apply Normalization rules in designing the tables in scope.

UNIT IV	TRANSACTIONS, CONCURRENCY CONTROL AND DATA STORAGE	9+9

Transaction Concepts – ACID Properties – Schedules based on Recoverability, Serializability – Concurrency Control – Need for Concurrency – Locking Protocols – Two Phase Locking – Transaction Recovery –Concepts – Deferred Update – Immediate Update.
Organization of Records in Files – Unordered, Ordered – Hashing Techniques – RAID – Ordered Indexes – Multilevel Indexes - B+ tree Index Files – B tree Index Files.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list
   a) Inventory Management for a EMart Grocery Shop
   b) Society Financial Management
   c) Cop Friendly App – Eseva
   d) Property Management – eMall
   e) Star Small and Medium Banking and Finance

Ability to showcase ACID Properties with sample queries with appropriate settings for the above scenario.

UNIT V	QUERY OPTIMIZATION AND ADVANCED DATABASES	9+9

Query Processing Overview – Algorithms for SELECT and JOIN operations – Query optimization using Heuristics.
Distributed Database Concepts – Design –Concurrency Control and Recovery – NOSQL Systems – Document-Based NOSQL Systems and MongoDB.
Explain Plan Statement – Parsing Output – Join Orders and Methods – Indexes - Standard Issues – Query Tuning - Explain Plan vs Explain Analyses.

List of Exercise/Experiments

Case Study using real life database applications anyone from the following list
   a) Inventory Management for a EMart Grocery Shop
   b) Society Financial Management
   c) Cop Friendly App – Eseva
   d) Property Management – eMall
   e) Star Small and Medium Banking and Finance

Build PL SQL / Stored Procedures for Complex Functionalities, ex EOD Batch Processing for calculating the EMI for Gold Loan for each eligible Customer.

TOTAL: 45 + 45 = 90 PERIODS

OUTCOMES:

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

CO1: Map ER model to Relational model to perform database design effectively.
CO2: Implement SQL and effective relational database design concepts.
CO3: Apply relational algebra, calculus and normalization techniques in database design.
CO4: Understand the concepts of transaction processing, concurrency control, recovery procedure and data storage techniques.
CO5: Evaluate and implement transaction processing, concurrency control mechanisms, and recovery procedures to maintain data integrity.
CO6: Analyze and optimize database queries and understand the features and applications of advanced and distributed database systems, including NoSQL.

TEXTBOOKS:

1. Elmasri R. and S. Navathe, "Fundamentals of Database Systems", Pearson Education, 7th Edition, 2016.
2. Abraham Silberschatz, Henry F.Korth, "Database System Concepts", Tata McGraw Hill , 7th Edition, 2021.

REFERENCES:

1. Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education, 2013.Raghu Ramakrishnan, Gehrke "Database Management Systems", MCGraw Hill, 3rd Edition 2014.
2. Plunkett T., B. Macdonald, "Oracle Big Data Hand Book" , McGraw Hill, First Edition, 2013
3. Gupta G K , "Database Management Systems" , Tata McGraw Hill Education Private Limited, New Delhi, 2011
4. C. J. Date, A. Kannan, S. Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2015.
5. Maqsood Alam, Aalok Muley, Chaitanya Kadaru, Ashok Joshi, Oracle NoSQL Database: Real-Time Big Data Management for the Enterprise, McGraw Hill Professional, 2013.
6. Thomas Connolly, Carolyn Begg, "Database Systems: A Practical Approach to Design, Implementation and Management", Pearson, 6th Edition, 2015.
7. Database Management System Part – 1
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview
8. Database Management System Part – 2
   https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012767300562919424 1_shared/overview
9. Online Resources:
10. https://infyspringboard.onwingspan.com/web/en/page/home

LIST OF EQUIPMENTS

Systems with MySQL, Eclipse or NetBeans

24AM301	ARTIFICIAL INTELLIGENCE AND DECISION MAKING	L	T	P	C
		2	0	2	3

OBJECTIVES:

- To implement various search strategies in AI.
- To learn about different problem-solving strategies using heuristic function.
- To learn about knowledge-based agents and first order logics.
- To understand knowledge representation and planning.
- To illustrate the uncertain knowledge representation.
- To apply decision making to solve real world problems.

UNIT I	PROBLEM SOLVING	6+6

Search Algorithms - Uninformed Search Strategies - Heuristic search strategies – Heuristic functions – Local Search and Optimization Problems – Local Search in Continuous spaces – Search with Nondeterministic actions – Search in Partially Observable Environments – Online Search Agents and Unknown Environments.

Lab Programs:

1. Implement basic search strategies – 8-Puzzle, 8 - Queens problem.
2. Implement Breadth First Search & Depth first Search Algorithm
3. Implement Water Jug problem.
4. Solve Tic-Tac-Toe problem.

UNIT II	CSP AND GAME SEARCH ALGORITHMS	6+6

Constraint Satisfaction Problems - Constraint propagation – Backtracking search for CSP – Local search for CSP – Structure of Problems - Game Theory - Optimal decisions in games – Alpha-beta search – Monte-Carlo search – Stochastic Games – Partially observable Games – Limitations of Game search Algorithms.

Lab Programs:

1. Implement A* and memory bounded A* algorithms.
2. Implement Minimax algorithm & Alpha-Beta pruning for game playing.
3. Solve Constraint Satisfaction Problems.

Mini Project – Chess/ Sudoku.

UNIT III	LOGICAL AGENTS	6+6

Knowledge-based agents – Wumpus World - Logic - Propositional logic – Propositional theorem proving – Propositional model checking – Agents based on propositional logic - First-Order Logic – Representation - Syntax and semantics – Using First-Order Logic - Knowledge Engineering in first-order logic – Propositional Vs First-Order Inference - Unification and First-Order Inference - Forward chaining – Backward chaining – Resolution.

Lab Programs:

1. Implement Unification algorithm for the given logic.

Implement forward chaining and backward chaining using Python.

UNIT IV	KNOWLEDGE REPRESENTATION AND PLANNING	6+6

Ontological engineering – Categories and objects – Events – Mental objects and modal logic – Reasoning systems for categories – Reasoning with default information - Classical planning – Algorithms for classical planning – Heuristics for planning – Hierarchical planning – Non-deterministic domains – Time, schedule, and Resources – Analysis.

Lab Programs:

1. Implementation of object detection.

Implement classical planning algorithms.

UNIT V	UNCERTAIN KNOWLEDGE AND REASONING	6+6

Quantifying Uncertainty – Probabilistic Reasoning – Probabilistic Reasoning over Time – Making Simple Decisions – Combining beliefs and desires under uncertainty – Utility Theory – Utility Functions – Multi-attribute Utility Functions – Decision Networks – Value of Information – Unknown Preferences.

Lab Programs:

1. Implement Bayesian networks and perform inferences.

Build a utility function using people choices with different features (multi-attribute utility function).

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Implement various search strategies.

CO2: Apply search strategies in problem solving and game playing using heuristic function.

CO3: Implement logical agents and first-order logic problems.

CO4: Apply problem-solving strategies with knowledge representation mechanism for solving hard problems.

CO5: Represent uncertain knowledge and build a decision network using a real-world scenario.

CO6: Implement AI algorithms to solve real-world problems.

TEXT BOOKS:

1. Peter Norvig and Stuart Russel, Artificial Intelligence: A Modern Approach, Pearson, 4th Edition, 2022.
2. Bratko, Prolog: Programming for Artificial Intelligence‖, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.

REFERENCES:

1. Elaine Rich, Kevin Knight and B. Nair, Artificial Intelligence 3rd Edition, McGraw Hill, 2017.
2. Melanie Mitchell, Artificial Intelligence: A Guide for Thinking Humans. Series: Pelican Books, 2020
3. Ernest Friedman-Hill, Jess in Action, Rule-Based Systems in Java, Manning Publications, 2003
4. Nils J. Nilsson, The Quest for Artificial Intelligence, Cambridge University Press, 2009.
5. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems,1st Edition by Patterson, Pearson, India, 2015.
6. NPTEL Courses:
    a. An Introduction to Artificial Intelligence - https://onlinecourses.nptel.ac.in/noc23_cs05/preview

Artificial Intelligence: Knowledge Representation And Reasoning - https://onlinecourses.nptel.ac.in/noc23_cs09/preview

LIST OF EQUIPMENTS:

Systems with C++/Python

24GE311	PRODUCT DEVELOPMENT LAB - 1 (Design Thinking) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Identify any gaps in the literature, such as open questions or conflicts in previous studies.
- Identify the need for additional research and areas of prior research to prevent duplication of effort.
- encourage the rapid and creative generation of diverse ideas within a limited timeframe, promoting innovation, critical thinking, and problem-solving in relation to a defined problem or research area.

DESCRIPTION

The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. The students learn to observe, empathize and ideate innovative solutions to real-world challenges. The student batches should identify a proposed work and will do literature review to develop prototype idea. Further at the end of the semester they will make a final presentation to exhibit the Identified literature gap.

LIST OF ACTIVITIES:

1. Conduct a literature review on the identified research area.
2. Identify the research gap from the literature review conducted.
3. Brain storming to be done to generate a large number of ideas within a specific timeframe.
4. Present the literature review conducted and identified research gap in relation to the research problem being investigated.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Identify, collect, and interpret relevant scholarly literature in the chosen research area.

CO2: Evaluate existing research to identify gaps and formulate precise research questions.

CO3: Apply brainstorming techniques to generate innovative and diverse research ideas.

CO4: Analyze reviewed literature and research gap to effectively present a research problem with clarity and rationale.

CO5: Develop skills in using evidence to create and present an engaging and critical argument.

24CS311	APTITUDE AND CODING SKILLS – I (Common to All Branches)	L	T	P	C
		0	0	3	1.5

OBJECTIVES:

The Course will enable learners to:

- Develop vocabulary for effective communication and reading skills.
- Build the logical reasoning and quantitative skills.
- Develop error correction and debugging skills in programming.

List of Exercises:

1. **English – Phase I**

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. **Logical Reasoning – Phase I**

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. **Quantitative Ability - Phase I**

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. **Automata Fix – Phase I**

Logical, Compilation and Code reuse

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop vocabulary for effective communication skills.

CO2: Build the logical reasoning enhance critical thinking.

CO3: Develop error correction and debugging skills in programming.

CO4: Apply programming skills to develop programs efficiently

CO5: Solve problems using quantitative skills

CO6: Develop effective reading and listening skills.

SEMESTER – IV

24MA401	PROBABILITY AND STATISTICS (Common to CSE, IT and AIML)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Provide the necessary basic concepts of random variables and introduce some standard distributions.
- Comprehend the concepts of joint distributions, marginal and conditional distributions.
- Test the hypothesis for small and large samples.
- Introduce the concepts of analysis of variances.
- Understand the concept of statistical quality control.

UNIT I ONE-DIMENSIONAL RANDOM VARIABLES 9+6

Basic probability, Independent events, Conditional probability (definition) - Random variable - Discrete and continuous random variables - Moments - Moment generating functions - Binomial, Poisson, Geometric, Uniform, Exponential and Normal distributions.

List of Exercises/Experiments using R Programming:

1. Finding conditional probability.
2. Finding mean, variance and standard deviation.

UNIT II TWO-DIMENSIONAL RANDOM VARIABLES 9+6

Joint distributions - Marginal and conditional distributions - Covariance - Correlation and linear regression - Transformation of random variables.

List of Exercises/Experiments using R Programming:

1. Finding marginal density functions for discrete random variables.
2. Calculating correlation and regression.

UNIT III TESTING OF HYPOTHESIS 9+6

Sampling distributions - Estimation of parameters - Statistical hypothesis - Large sample tests based on Normal distribution for single mean and difference of means - Tests based on t and F distributions for mean and variance - Chi-square test- Contingency table (test for independent) - Goodness of fit.

List of Exercises/Experiments using R Programming:

1. Testing of hypothesis for given data using z - test.
2. Testing of hypothesis for given data using t - test.

UNIT IV DESIGN OF EXPERIMENTS 9+6

One way and Two-way classifications - Completely randomized design - Randomized block design - Latin square design.

List of Exercises/Experiments using R Programming:

1. Perform one-way ANOVA test for the given data.
2. Perform two-way ANOVA test for the given data.

UNIT V STATISTICAL QUALITY CONTROL 9+6

Control charts for measurements ($\bar{X}$ and R charts) - Control charts for attributes (p, c and np charts) - Tolerance limits.

List of Exercises/Experiments using R Programming:

1. Interpret the results for $\bar{X}$- Chart for variable data.
2. Interpret the results for R-Chart for variable data.

TOTAL: 45+30=75 PERIODS

OUTCOMES:

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

CO1: Compute the statistical measures of standard distributions.
CO2: Apply joint, marginal and conditional distributions to solve practical problems
CO3: Determine the correlation and regression for two dimensional random variables
CO4: Employ the concept of testing of hypothesis to solve real life problems.
CO5: Apply the concept of analysis of variance for various experimental designs.
CO6: Prepare the control charts for variables and attributes for analyzing the data.

TEXT BOOKS:

1. R.A. Johnson, I. Miller and J. Freund, "Miller and Freund's Probability and Statistics for Engineers", Pearson Education, Asia, 9th Edition, 2023.
2. J.S. Milton and J.C. Arnold, "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2019.

REFERENCES:

1. J. L. Devore, "Probability and Statistics for Engineering and the Sciences", Cengage Learning, New Delhi, 9th Edition, Reprint 2020.
2. S. M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 6th Edition, Elsevier, 2020.
3. M. R. Spiegel, J. Schiller and R.A. Srinivasan, "Schaum's Outline of Theory and Problems of Probability and Statistics", Tata McGraw Hill, 4th Edition, 2013.
4. R. E. Walpole, R. H. Myers, S.L. Myers and K. Ye, "Probability and Statistics for Engineers and Scientists". Pearson Education, Asia, 9th Edition, Reprint 2021.

24CS402	DESIGN AND ANALYSIS OF ALGORITHMS (Common to CSE, IT and AIML)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Critically analyse the efficiency of alternative algorithmic solutions for the same problem
- Illustrate brute force and divide and conquer design techniques.
- Explain dynamic programming for solving various problems.
- Apply greedy technique and iterative improvement technique to solve optimization problems
- Examine the limitations of algorithmic power and handling it in different problems.

UNIT I	INTRODUCTION	9+6

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving –Fundamentals of the Analysis of Algorithmic Efficiency – Asymptotic Notations and their properties. Analysis Framework – Mathematical analysis for Recursive and Non-recursive algorithms

List of Exercise/Experiments:

1. Perform the recursive algorithm analysis.
2. Perform the non-recursive algorithm analysis.

UNIT II	BRUTE FORCE AND DIVIDE AND CONQUER	9+6

Brute Force - String Matching - Exhaustive Search - Knapsack Problem - Divide and Conquer Methodology – Binary Search – Merge sort – Quick sort - Multiplication of Large Integers – Closest-Pair and Convex Hull Problems - Transform and Conquer Method: Heap Sort

List of Exercise/Experiments:

1. Write a program to search an element using binary search
2. Write a program to sort the elements using merge sort and find time complexity.

UNIT III	DYNAMIC PROGRAMMING	9+6

Dynamic programming – Principle of optimality – Floyd‘s algorithm – Multi stage graph - Optimal Binary Search Trees - Longest common subsequence - Matrix-chain multiplication – Travelling Salesperson Problem – Knapsack Problem and Memory functions.

List of Exercise/Experiments:

1. Solve Floyd’s algorithm
2. Write a program to find the longest common subsequence

UNIT IV	GREEDY TECHNIQUE AND ITERATIVE IMPROVEMENT	9+6

Greedy Technique – Prim‘s algorithm and Kruskal’s Algorithm – Huffman Trees. The Maximum-Flow Problem – Maximum Matching in Bipartite Graphs - The Stable marriage Problem

List of Exercise/Experiments:

1. Write a program to find minimum spanning tree using Prim’s algorithm
2. Implement Kruskal’s algorithm to find minimum spanning tree

UNIT V	BACKTRACKING AND BRANCH AND BOUND	9+6

P, NP NP- Complete and NP Hard Problems. Backtracking – N-Queen problem - Subset Sum Problem. Branch and Bound – LIFO Search and FIFO search - Assignment problem – Knapsack Problem - Approximation Algorithms for NP-Hard Problems – Travelling Salesman problem

List of Exercise/Experiments:

1. Write a program to implement sum of subset problem.
   Solve knapsack problem using branch and bound technique

TOTAL: 45+30=75 PERIODS

OUTCOMES:

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

CO1: Understand the different algorithm design paradigms.

CO2: Design algorithms for real world problems using algorithmic design techniques.

CO3: Analyse the efficiency of simple recursive and non-recursive algorithms.

CO4: Analyse the algorithm's worst, best and average case behaviour in terms of time and space.

CO5: Understand the approximation algorithms for solving NP Hard problems

CO6: Solve the problems by selecting suitable algorithmic design techniques.

TEXT BOOKS:

1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2019.

REFERENCES:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, Third Edition, PHI Learning Private Limited, 2012.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford university press, 2014.
3. http://nptel.ac.in/

LIST OF EQUIPMENTS:

1. Standalone PC with C/C++/Java

24AM401	MACHINE LEARNING	L	T	P	C
		2	0	2	3

OBJECTIVES:

- To discuss the basics of Machine Learning and model evaluation.
- To study dimensionality reduction techniques.
- To understand the various classification algorithms.
- To elaborate on unsupervised learning techniques.
- To discuss the basics of neural networks and various types of learning.

UNIT I	INTRODUCTION	6+6

Machine Learning – Types – Applications – Preparing to Model – Activities – Data – Exploring structure of Data – Data Quality and Remediation – Data Pre-processing – Modelling and Evaluation: Selecting a Model -Training a Model – Model representation and Interpretability – Evaluating Performance of a Model – Improving Performance.

Lab Programs:

1. Implementation of Candidate Elimination algorithm
2. Implementation of ML model evaluation techniques (R-Squared/Adjusted R-Squared/Mean Absolute Error/Mean Squared Error)
3. Implementation of ML model evaluation techniques (Confusion Matrix/F1 Score/AUC-ROC Curve)

UNIT II	FEATURE ENGINEERING AND DIMENSIONALITY REDUCTION	6+6

Feature Engineering – Feature Transformation – Feature Subset Selection - Principle Component Analysis – Feature Embedding – Factor Analysis – Singular value decomposition and Matrix Factorization – Multidimensional scaling – Linear Discriminant Analysis – Canonical Correlation Analysis – Isomap – Locally linear Embedding – Laplacian Eigenmaps.

Lab Programs:

1. Write python code to identify feature co-relations (PCA)
2. Interpret Canonical Covariates with Heatmap
3. Feature Engineering is the way of extracting features from data and transforming them into formats that are suitable for Machine Learning algorithms. Implement python code for Feature Selection/ Feature Transformation/ Feature Extraction.

Mini Project – Feature Subset Selection

UNIT III	SUPERVISED LEARNING	6+6

Linear Regression -Relation between two variables – Steps – Evaluation – Logistic Regression – Decision Tree – Algorithms – Construction – Classification using Decision Tree – Issues – Rule-based Classification – Pruning the Rule Set – Support Vector Machines – Linear SVM – Optimal Hyperplane – Radial Basis Functions – Naïve Bayes Classifier – Bayesian Belief Networks.

Lab Programs:

1. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select the appropriate data set for your experiment and draw graphs.
2. Implement and demonstrate the working of the decision tree-based ID3 algorithm

Build a Simple Support Vector Machines using a data set

UNIT IV	UNSUPERVISED LEARNING	6+6

Clustering – Types – Applications - Partitioning Methods – K-means Algorithm – K-Medoids – Hierarchical methods – Density based methods DBSCAN – Finding patterns using Association Rules – Hidden Markov Model.

Lab Programs:

1. Implement a K-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions
2. Implement market basket analysis using association rules

Mini Project using Clustering analysis.

UNIT V	NEURAL NETWORKS AND TYPES OF LEARNING	6+6

Biological Neuron – Artificial Neuron – Types of Activation function – Implementations of ANN – Architectures of Neural Networks – Learning Process in ANN – Back propagation – Deep Learning – Representation Learning – Active Learning – Instance based Learning – Association Rule Learning – Ensemble Learning Algorithm: Bagging – Boosting – GBM – XGBoost – Regularization Algorithm- Reinforcement Learning – Elements.

Lab Programs:

1. Build an ANN by implementing the Single-layer Perceptron. Test it using appropriate data sets.
2. Implement Multi-layer Perceptron and test the same using appropriate data sets.
3. Build a RBF Network to calculate the fitness function with five neurons.
4. Mini Project – Face recognition.

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Explain the basics of Machine Learning and model evaluation.
CO2: Study dimensionality reduction techniques.
CO3: Understand and implement various classification algorithms.
CO4: Understand and implement various unsupervised learning techniques.
CO5: Build Neural Networks and understand the different types of learning.
CO6: Use Machine Learning Algorithms to build applications.

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, "Machine Learning", Pearson, 2019. (Unit 1 – chap 1,2,3/ Unit 2 – Chap 4 / Unit 4 – 9 / Unit 5 – Chap 10, 11)
2. Ethem Alpaydin, "Introduction to Machine Learning, Adaptive Computation and Machine Learning Series", Third Edition, MIT Press, 2014. (Unit 2 – Chap 6 / Unit 4 – chap 8.2.3/ Unit 5 – Chap 18)

REFERENCES:

1. Anuradha Srinivasaraghavan,Vincy Joseph, "Machine Learning", First Edition, Wiley, 2019.(Unit 3 – Chap 7,8,9,10,11 / Unit 4 – 13, 11.4, 11.5,12)
2. Sebastian Raschka, Python Machine Learning, Packt Publications, Second Edition, 2017.
3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly Publications, Third Edition, 2022.
4. Peter Harrington, "Machine Learning in Action", Manning Publications, 2012.
5. Stephen Marsland, "Machine Learning – An Algorithmic Perspective", Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
6. Tom M Mitchell, "Machine Learning", First Edition, McGraw Hill Education, 2013.
7. Bishop C.M., Pattern Recognition And Machine Learning, Springer, 2009.
8. NPTEL Courses:
9. Introduction to Machine Learning - https://onlinecourses.nptel.ac.in/noc23_cs18/preview

LIST OF EQUIPMENTS:

Systems with Anaconda, Jupyter Notebook, Python, Pytorch, scikit-learn, Tensorflow, Colab

24AM402	NEURAL NETWORKS	L	T	P	C
		2	0	2	3

OBJECTIVES:

- To understand the biological neural network and to model equivalent neuron models.
- To understand the architecture, learning algorithms.
- To know the issues of various feed forward and feedback neural networks.
- To gain deep insight about Boltzmann Machine Learning
- To explore Autoencoders and Hopfield Nets

UNIT I	INTRODUCTION	6+6

A Neural Network, Human Brain, Models of a Neuron, Neural Networks viewed as Directed Graphs, Network Architectures, Knowledge Representation, Artificial Intelligence and Neural Networks Learning Process: Error Correction Learning, Memory Based Learning, Hebbian Learning, Competitive, Boltzmann Learning, Credit Assignment Problem, Memory, Adaption, Statistical Nature of the Learning Process. A simple example of learning – Three types of Learning – Types of Neural Network Architectures

Lab Programs:

Study of JAX and its installation
Perform matrix operations.
Plot multiple curves in single plot.
Plot Activation function used in neural network
Create a simple neural network

UNIT II	PERCEPTRONS	6+6

Adaptive Filtering Problem, Unconstrained Organization Techniques, Linear Least Square Filters, Least Mean Square Algorithm, Learning Curves, Learning Rate Annealing Techniques, Perceptron –Convergence Theorem, Relation Between Perceptron and Bayes Classifier for a Gaussian Environment – A geometrical view of Perceptrons – What perceptrons can't do Multilayer Perceptron: Back Propagation Algorithm XOR Problem, Heuristics, Output Representation and Decision Rule, Computer Experiment, Feature Detection

Lab Programs:

Create a Perceptron.
Pattern Classification using Perceptron network.
Build a neural network by implementing the Single-layer Perceptron. Test it using appropriate data sets.

UNIT III	BACK PROPAGATION	6+6

Learning the weights of a linear neuron-error surface – learning weights of logistic output neuron-Back Propagation and Differentiation, Hessian Matrix, Generalization, Cross Validation, Network Pruning Techniques, Virtues and Limitations of Back Propagation Learning, Accelerated Convergence, Supervised Learning

Lab Programs:

Create a Back Propagation Feed-forward neural network.
Implement and train a Bayesian Neural network.
Compute and visualize the Hessian for a 2D weight system.

UNIT IV	BOLTZMANN MACHINE LEARNING	6+6

How a Boltzmann machine models data - Restricted Boltzmann machine- example of RBM learning-Collaborative filtering-learning layers of features by stacking RBMs.

Lab Programs:

Model real valued data with RBM.
Demonstrate looking for patterns in gene expression profiles in baker's yeast.

UNIT V	AUTOENCODERS AND HOPFIELD NETS	6+6

From PCA to autoencoders-Deep autoencoders-document retrieval- semantic hashing – learning binary codes for image retrieval- shallow autoencoders Hopfield Network – Hopfield Models- Hopfield nets with hidden units

Lab Programs:

Design a Hopfield Network which stores 4 vectors
Image retrieval
Digit Retrieval using Autoencoders and Hopfield Networks.

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Understand the similarity of Biological networks and Neural networks
CO2: Perform the training of neural networks using various learning rules.
CO3: Understand the concepts of forward and backward propagations.
CO4: Analyze the architecture and learning algorithms of Boltzmann Machines and their applications in deep learning.
CO5: Design and train Hopfield networks for associative memory tasks and evaluate their performance.
CO6: Develop and experiment with autoencoders (shallow and deep) for feature extraction and data compression.

TEXT BOOKS:

1. Neural Networks a Comprehensive Foundations, Simon S Haykin, PHI Ed
2. Introduction to Artificial Neural Systems Jacek M. Zurada, JAICO Publishing House Ed. 2006.
3. Christopher M Bishop, Pattern Recognition and Machine Learning. Springer. 2011.
4. Geoffrey Hintonand Terrence J. Sejnowski,Unsupervised Learning: Foundations of Neural Computation.

REFERENCES:

1. Neural Networks for Machine Learning – Geoffrey E. Hinton, UoFT https://www.youtube.com/playlist?list=PLLssT5z_DsK_gyrQ_biidwvPYCRNGI3iv
2. Neural Networks in Computer Intelligence, Li Min Fu TMH 2003
3. Neural Networks -James A Freeman David M S Kapura Pearson Ed., 2004.
4. Artificial Neural Networks – B. Vegnanarayana Prentice Hall of India P Ltd 2005

24IT402	WEB DEVELOPMENT FRAMEWORKS	L	T	P	C
		3	0	3	4.5

COURSE OBJECTIVES:

The Course will enable the learners:

- To understand web semantics and related tools and framework
- Able to get hands on latest JS based web frameworks
- To develop a scalable and responsive web application
- To develop an industry ready application web enterprise feature

UNIT I	ADVANCED TYPESCRIPT	9+9

Introduction to HTML5 and CSS3, Media Queries, JS, DOM, BootStrap, Variables, Loops, Operators, Scope, Hoisting, Arrays, Spread, REST, DeStructuring

Introduction, Advantage of Using TS over JS, and where to Use and not to use TS - Understanding the Compiler (Transpiler), and its options, Scope of TS - Variable Scopes, Static Data Types - String, Number,Interface, Date - Union, Tuple, Undefined Data Types, Unknown vs any vs Never - Object Oriented , Arrow Funcions - Types, KeyOf, TypeOf, InstanceOf ,Narrowing, Conditional Types - Generics,Enum ,Required / Partial / Optional - Arrays, Modules ,Async Processing w Call backs ,Type Inference, Type Compatability, Utility Type - Unit Testing,TSLint

List of Exercise/Experiments

1. Create a TS Object for Bank Account (w attributes like à customer name, account type, balance, data of creation, bank name, branch name, pan card number). Using JS Object keyword, try to perform following activities
    - List down all the entries of the bank object
    - Check the existence of a key
    - If key found, get the value for the key
2. Spread Operator
    - Merge Customer and Account Arrays
    - Update the Customer Object with the new values
    - Develop a function that takes an Spread Argument and calculates total balance.

UNIT II	INTRODUCTION TO REACTJS	9+9

Introduction to React - ES6 Features,What is React?,React Features - Setting up React Development Environment:- Node.js and npm installation,Create React App,Project structure - JSX (JavaScript XML):- What is JSX?,JSX Syntax and Rules,JSX Expressions - Components in React:- Functional Components,Class Components,Props and PropTypes - State and Lifecycle :- State and setState,Lifecycle Methods,Mounting, Updating, and Unmounting,Handling Events in React - Event Handling in React :- Synthetic Events,Event Binding, - Conditional Rendering:-If-else Statements,Ternary Operator,Logical && Operator
Lists and Keys:-Rendering Lists,Keys and Reconciliation,Extracting Components.

List of Exercise/Experiments

1. A leading bank from APAC wants to modernize their banking services and decided to build a online multi channel mobile ecommerce platform. As part of the drive, starting building following feature set in a staggered model emplying ReactJS as front end library and associated libs from React eco system. Feature to be implemented are

- User Login Page
- Account Summary
- Funds Transfer (within bank and outside bank)
- Recurring and Fixed deposits
- Letter of Credit
- Salary or 3rd Party Payment

2. Unit 2 Scope – Project Setup, Web App Layout Completion using BootStrap or Tailwind, Login Page Implementation, Landing Page Implementation, Authentication and Authorization Implementation.

UNIT III	REACTJS COMPONENTS	9+9

Forms and Controlled Components :- Form Handling in React, Controlled Components, Uncontrolled Components - Basic Hooks :- useState,useRef,useEffect, - Routing in React:- Introduction to React Router,Route, Link, and Switch Components, Route Parameters,useNavigate,useParams, - REST API – Axios GET/PUT/Delete/Remove, Interceptor, Headers, Authorization Token, Promise and Observables (via rxjs)

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
    - Unit 3 Scope – Forms and Validation (React Form Validation), Integration of Back End Apis via Axios,API Security Implementaiton, Routes and Navigation with Priviate Routes, Usage of useEffect, UseContext hooks

UNIT IV	REACT PRO TOOLKIT: ERROR MANAGEMENT, ABSTRACTIONS AND DATA HANDLING	9+9

Error Handling :- Error Boundaries,componentDidCatch,Error Handling Strategies
Higher-Order Components (HOCs) :- What are HOCs?,Creating and Using HOCs,HOCs vs Render Props,Code Splitting and Lazy Loading,Server-Side Rendering - Data Fetching with React Query :- React Query,Introduction to React Query,Query Keys and Query Functions, Query Invalidation and Refetching.

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
    - Unit 4 Scope – Completion of Remaining Modules, Error Handling, HOC and AUX implementation, Lazy loaded components for improved performance
2. Extend the Project developed in previous section with newly learnt concepts

UNIT V	REACT UNDER THE HOOD: TESTING, CONTEXT API, AND REDUX	9+9

Testing React Components :- Introduction to Testing,Jest Framework,React Testing Library
Context API :- Creating Context,Providing and Consuming Context,useContext Hook
Redux Overview :- What is Redux?,Redux Principles,Single Source of Truth - Redux Actions and Reducers :- Redux Actions,Redux Reducers,Combining Reducers.

List of Exercise/Experiments

1. Extend the Project developed in previous section with newly learnt concepts
    - Unit 5 Scope – Unit Testing using JEST, Redux implementation for state management.

Business Use Case Implementations

1. Student Management System
2. Retail Bank System
3. eCommerce System
4. Student LMS Management System

TOTAL: 45+45=90 PERIODS

COURSE OUTCOMES:

After completing the course, students will have the ability to

CO1. Understand and apply modern web technologies including HTML5, CSS3, JavaScript, and advanced TypeScript concepts to build dynamic web components.

CO2. Develop responsive and modular front-end applications using ReactJS, including component creation, state management, and routing.

CO3. Implement advanced React features like hooks (useState, useEffect, useRef), React Router, and REST API integration using Axios for dynamic content handling.

CO4. Utilize higher-order components (HOCs), lazy loading, and server-side rendering to optimize and abstract React applications.

CO5. Perform unit testing using Jest and RTL, and manage global application state efficiently using Context API and Redux.

CO6. Design and deliver scalable and real-world enterprise web applications with complete user interface flow, security, and error handling.

TEXTBOOKS:

1. David Flanagan, Javascript The Definitive Guide, Paperback, 7th Edition, 2020.
2. David Choi ,Full-Stack React, TypeScript, and Node: Build cloud-ready web applications using React 17 with Hooks and GraphQL Paperback – Import, 18 December 2020.
3. Mehul Mohan, Advanced Web Development with React Paperback – 1 January 2020.

E-RESOURCES:

1. Parental Website - https://reactjs.org/
2. The Road to Learn React: Your journey to master plain yet pragmatic React.js by Robin Wieruch
3. Learning React: Functional Web Development with React and Redux by Alex Banks and Eve Porcello
4. Learning React by KirupaChinnathambi
5. "React Up & Running" by StoyanStefanov
6. https://www.edureka.co/reactjs-redux-certification-training
7. CodePen
8. CodeSandbox (Preferred)

9. Stackblitz

LIST OF EQUIPMENTS:

- NodeJS (v22.11.2)
- Github as code repository
- Visual studio code as IDE
- RTL as unit testing framework
- Responsive design w bootstrap
- ReactJS installation (v17)
- Chrome / FIreFox Browsers (latest)
- Responsive using Media Queries & Bootstrap Material&Antdesign
- Design based Apps

24GE411	PRODUCT DEVELOPMENT LAB - 2 (Functional Design) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- To accurately capture the expectations, needs, and constraints of stakeholders to ensure that the system being developed aligns with user goals and real-world usage scenarios.
- To visually represent the system's structure and functional relationships, enabling clearer communication, easier analysis, and better planning of system architecture and workflow.
- To create a precise and comprehensive reference document that guides development, ensures clarity, and serves as a basis for verification, validation, and future system enhancements.

DESCRIPTION

The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. Further at the end of the semester they will make a final presentation to exhibit the functional design and the process to develop a product.

LIST OF ACTIVITIES:

1. Understand and document functional requirements based on stakeholder needs.
2. Map out how functions interact and relate to each other using functional block diagrams or flowcharts.
3. Write detailed specification describing each function's role, behavior, constraints, and expected outcomes.
4. Check if defined functions meet the intended requirements and can be validated/tested.
5. Present the design concepts carried out for the identified research gap in relation to the research problem being investigated.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Interpret stakeholder needs and document comprehensive functional requirements for the proposed system.

CO2: Develop functional block diagrams or flowcharts to represent system interactions and functional relationships.

CO3: Analyze functional specifications that define roles, behaviors, constraints, and performance expectations for each function.

CO4: Evaluate the defined functional model through verification and validation techniques to ensure alignment with original requirements.

CO5: Analyze and present functional design solutions aligned with the identified research problem and gap.

LIST OF EQUIPMENTS FOR A BATCH OF 30 STUDENTS

S. No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos
6	Solder Stations	5 Sets

7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Modelling Software(open source)	30 Nos

24CS411	APTITUDE AND CODING SKILLS – II (Common to All Branches)	L	T	P	C
		0	0	3	1.5

OBJECTIVES:

The Course will enable learners to:

- Develop advanced vocabulary for effective communication and reading skills.
- Build an enhanced level of logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.
- To apply data structures and algorithms in problem solving.

List of Exercises:

1. **English – Phase II**

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. **Logical Reasoning – Phase II**

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. **Quantitative Ability - Phase II**

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. **Automata Fix – Phase II**

Logical, Compilation and Code reuse

5. **Automata - Phase II**

Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop advanced vocabulary for effective communication skills.
CO2: Build an enhanced level of logical reasoning and quantitative skills.
CO3: Develop error correction and debugging skills in programming.
CO4: Apply data structures and algorithms in problem solving.
CO5: Develop advanced vocabulary for effective reading skills
CO6: Apply advanced algorithm design techniques to develop programs

SEMESTER V

24AM501	DEEP LEARNING (Lab Integrated)	L	T	P	C
		3	0	2	4

OBJECTIVES:

- To understand the basics of deep neural networks.
- To implement deep learning models.
- To elaborate CNN and RNN architectures of deep neural networks.
- To familiarize autoencoders in neural networks.
- To learn about the deep generative models.
- To apply Deep Learning to solve real-world problems.

UNIT I **DEEP NETWORKS** **9+6**

Challenges motivating deep learning - Deep feedforward networks - Learning XOR - Gradient based learning - Hidden Units – Architecture Design – Back Propagation – Regularization – Parameter Norm Penalties – Constrained Optimization – Under-Constrained Problems – Dataset Augmentation – Noise Robustness – Semi-Supervised Learning – Multi-Task Learning – Early Stopping – Parameter Tying and Sharing – Bagging and Other Ensemble methods – Dropout – Adversarial Training.

List of Exercises:

1. Implement a simple feed-forward neural network.
    a. Create a basic network
    b. Analyze performance by varying the batch size, number of hidden layers, learning rate.
    c. Create a confusion matrix to validate the performance of your model.
    d. Visualize a neural network.
2. Solve XOR problem using Multi Layer Perceptron.

UNIT II **OPTIMIZATION FOR TRAINING DEEP MODELS** **9+6**

Pure optimization – Challenges – Basic Algorithms – Parameter initialization Strategies – Algorithms with Adaptive Learning Rates – Approximate Second-Order methods – Optimization Strategies and Meta Algorithms.

List of Exercises:

1. Implement Stochastic Gradient Descent Algorithm.
2. Implement Gradient Descent with AdaGrad.

UNIT III **CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS** **9+6**

Convolution Operation – motivation – Pooling – Infinitely Strong prior – Variants – Structured Output – Data Types – Efficient Convolutional Algorithms – Random or Unsupervised features – Neuroscientific Basis - Deep Learning – Sequence Modelling - Computational Graphs - RNN - Bidirectional RNN – Encoder-Decoder - Sequence to Sequence RNN - Deep Recurrent Networks - Recursive Neural Networks - Long Term Dependencies; Leaky Units - Strategies for multiple time scales – LSTM and Gated RNNs - Optimization for Long Term Dependencies.

List of Exercises:

1. Implement a Recurrent Neural Networks (RNN) and process any sequential data such as characters, words or video frames.
2. Implement RNN with Long Short Term Networks (LTSM).

UNIT IV **AUTOENCODERS** **9+6**

Autoencoders: Undercomplete autoencoders - Regularized autoencoders – Power, Layer Size and Depth - Stochastic encoders and decoders – Denoising Autoencoders - Learning with autoencoders – contractive Autoencoders – Applications of autoencoders.

List of Exercises:

1. Implement different types of autoencoders.

UNIT V **DEEP GENERATIVE MODELS** **9+6**

Boltzmann Machine – Restricted Boltzmann Machine – Deep Belief Networks – Deep Boltzmann Machines - Boltzmann Machines for Real-Valued Data – Convolutional Boltzmann Machines - Boltzmann Machine for Structured or Sequential Outputs – Directed Generative Nets – Evaluating

Generative Models.

List of Exercises:

1. Solve a real world problem using CBM.

TOTAL: 45 + 30 = 75 PERIODS

OUTCOMES:

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

CO1: Demonstrate the basics of deep neural networks to solve real world problems**.**
CO2: Implement deep learning models.
CO3: Elaborate CNN and RNN architectures of deep neural networks.
CO4: Use autoencoders in neural networks.
CO5: Illustrate the various deep generative models.
CO6: Apply deep generative models to solve real world problems.

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016.

REFERENCES:

1. Charu C. Aggarwal, "Neural Networks and Deep Learning: A Textbook", Springer International Publishing, 2018.
2. Yoav Goldberg, "Neural Network Methods for Natural Language Processing", Synthesis Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
3. Francois Chollet, "Deep Learning with Python", Manning Publications Co, 2018.
4. Josh Patterson, Adam Gibson, "Deep Learning: A Practitioner's Approach", O'Reilly Media, 2017.
5. Navin Kumar Manaswi, "Deep Learning with Applications Using Python", Apress, 2018.
6. Richard O. Duda, Peter E. Hart, David G. Stork, "Pattern Classification", John Wiley & Sons Inc., 2007.

24AM502	GENERATIVE AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Understand the foundational concepts and applications of Generative AI.
- Learn the working principles of generative models used for text, image, audio, and video generation.
- Apply Generative AI techniques for multimodal content generation.
- Understand speech synthesis and audio generation using Generative AI models.
- Evaluate and deploy Generative AI systems responsibly using standard evaluation metrics.

UNIT I	FOUNDATIONS OF GENERATIVE AI	9

Introduction to Generative AI – Types of Generative AI Models – Applications of Generative AI – Tokenization Techniques – Word Embeddings and Vector Representations – Attention Mechanism – Transformer Fundamentals – Prompt Engineering – Zero-shot and Few-shot Learning – Context Learning – Multimodal Generative AI – Ethical Considerations in Generative AI.

UNIT II	GENERATIVE AI CONCEPTS	9

Generative Models – Autoregressive Models – Diffusion Models – Latent Space Representation – Transfer Learning in Generative AI – Fine Tuning Concepts – Reinforcement Learning Concepts in Generative AI – Retrieval Augmented Generation (RAG) – Conversational AI – AI Agents – Hallucination in Generative AI – Ethical and Responsible AI Concepts.

UNIT III	IMAGE AND VIDEO GENERATION	9

Text-to-Image Generation – Diffusion Models – Stable Diffusion – DALL·E – Midjourney Concepts – Image Editing and Inpainting – ControlNet – Video Generation Models – Generative AI for Creative Applications – Ethical Issues in Synthetic Media – Deepfake Detection and Prevention.

UNIT IV	SPEECH SYNTHESIS AND AUDIO GENERATION	9

Introduction to Speech Synthesis – Text-to-Speech (TTS) Pipeline – WaveNet – Tacotron and Tacotron 2 – FastSpeech – Neural Vocoders – HiFi-GAN – Diffusion-Based Audio Generation – AudioLDM – Music Generation with Generative Models – Voice Cloning and Style Transfer.

UNIT V	GENERATIVE AI DEPLOYMENT AND EVALUATION	9

Evaluation Challenges in Generative AI – Inception Score (IS) – Frechet Inception Distance (FID) – CLIP Score – BLEU and ROUGE Metrics – Human Evaluation Protocols – Responsible Generative AI – Bias and Fairness – Content Safety and Watermarking – AI Governance – Deployment Considerations – Generative AI Applications in Healthcare, Education, Finance, and Software Development.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the foundational concepts, architectures, and applications of Generative AI.

CO2: Apply Generative AI concepts such as prompt engineering, retrieval augmentation, and conversational AI techniques.

CO3: Develop image and video generation applications using diffusion-based Generative AI models.

CO4: Build speech synthesis and audio generation systems using modern Generative AI techniques.

CO5: Evaluate and apply Generative AI systems responsibly using standard metrics and ethical practices.

TEXT BOOKS:

1. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023.
2. Li Deng and Dong Yu, Deep Learning: Methods and Applications, NOW Publishers, 2014. (Chapters on Speech Synthesis and Audio Modeling)

REFERENCES:

1. Martin Heusel et al., GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium (FID Paper), NeurIPS, 2017. Available: https://arxiv.org/abs/1706.08500
2. Aaron van den Oord et al., WaveNet: A Generative Model for Raw Audio, DeepMind, 2016. Available: https://arxiv.org/abs/1609.03499
3. Jonathan Shen et al., Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions (Tacotron 2), ICASSP, 2018. Available: https://arxiv.org/abs/1712.05884
4. Alec Radford et al., CLIP: Learning Transferable Visual Models from Natural Language Supervision, OpenAI, 2021. Available: https://arxiv.org/abs/2103.00020
5. Yi Ren et al., FastSpeech: Fast, Robust and Controllable Text to Speech, NeurIPS, 2019. Available: https://arxiv.org/abs/1905.09263
6. Jungil Kong et al., HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis, NeurIPS, 2020. Available: https://arxiv.org/abs/2010.05646
7. Haohe Liu et al., AudioLDM: Text-to-Audio Generation with Latent Diffusion Models, ICML, 2023. Available: https://arxiv.org/abs/2301.12503
8. Sercan O. Arik et al., Neural Voice Cloning with a Few Samples, NeurIPS, 2018. Available: https://arxiv.org/abs/1802.06006
9. Kishore Papineni et al., BLEU: a Method for Automatic Evaluation of Machine Translation, ACL, 2002; Chin-Yew Lin, ROUGE: A Package for Automatic Evaluation of Summaries, ACL Workshop, 2004.
10. John Kirchenbauer et al., A Watermark for Large Language Models, ICML, 2023. Available: https://arxiv.org/abs/2301.10226

24CS304	OPERATING SYSTEMS (Lab Integrated)	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- Explain the basic concepts of operating systems and process.
- Discuss threads and analyse various CPU scheduling algorithms.
- Describe the concept of process synchronization and deadlocks.
- Analyse various memory management schemes.
- Describe I/O management and file systems.

UNIT I	INTRODUCTION TO OPERATING SYSTEMS AND PROCESSES	6+6

Introduction: Computer system organization - architecture – Resource management - Protection and Security – Virtualization - Operating System Structures: Services - User and Operating-System Interface - System Calls - System Services - Design and Implementation - Building and Booting an Operating System – Processes: Process Concept - Process Scheduling - Operations on Processes – Inter process Communication - IPC in Shared-Memory Systems - IPC in Message-Passing Systems

List of Exercise/Experiments:

1. Basic Unix file system commands such as ls, cd, mkdir, rmdir, cp, rm, mv, more, lpr, man, grep, sed, etc..
2. Programs using Shell Programming.
3. Implementation of Unix System Calls.
4. Implementation of IPC using message queue
   a. Get the input data (integer value) from a process called sender
   b. Use Message Queue to transfer this data from sender to receiver process
   c. The receiver does the prime number checking on the received data
   d. Communicate the verified/status result from receiver to sender process, this status should be displayed in the Sender process.

Note: Simultaneously execute two or more processes. Don't do it as a single process

UNIT II	THREADS AND CPU SCHEDULING	6+6

Threads & Concurrency: Overview - Multicore Programming - Multithreading Models - Thread Libraries - Implicit Threading - Threading Issues - CPU Scheduling: Basic Concepts – Scheduling Criteria - Scheduling Algorithms - Thread Scheduling - Multi-Processor Scheduling - Real-Time CPU Scheduling

List of Exercise/Experiments:

1. Write a program to implement the following actions using pthreads
   a. Create a thread in a program and called Parent thread, this parent thread creates another thread (Child thread) to print out the numbers from 1 to 20. The Parent thread waits till the child thread finishes
   b. Create a thread in the main program, this program passes the 'count' as arguments to that thread function and this created thread function has to print your name 'count' times.
2. Write C programs to implement the various CPU Scheduling Algorithms.

UNIT III	PROCESS SYNCHRONISATION AND DEADLOCKS	6+6

Process Synchronization: The critical-section problem – Peterson's Solution, Synchronization hardware, Mutex locks, Semaphores, monitors - Classic problems of synchronization: Bounded Buffer Problem - Reader's & Writer Problem, Dinning Philosopher Problem. Deadlock: System model - Deadlock characterization, Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock.

List of Exercise/Experiments:

1. Process Synchronization using Semaphores. A shared data has to be accessed by two categories of processes namely A and B. Satisfy the following constraints to access the data without any data loss.

a. When a process A1 is accessing the database another process of the same category is permitted.
b. When a process B1 is accessing the database neither process A1 nor another 74 process B2 is permitted.
c. When a process A1 is accessing the database process B1 should not be allowed to access the database. Write appropriate code for both A and B satisfying all the above constraints using semaphores.

Note: The time-stamp for accessing is approximately 10 sec.

2. Bankers Algorithm for Deadlock Avoidance

UNIT IV	MEMORY MANAGEMENT	6+6

Memory Management: Contiguous Memory Allocation - Paging - Structure of the Page Table – Swapping - Virtual Memory: Demand Paging – Copy-on write – Page Replacement – Allocation of frames – Thrashing – Memory Compression

List of Exercise/Experiments:

1. Analysis and Simulation of Memory Allocation and Management Techniques
   i. First Fit ii. Best Fit iii. Worst Fit
2. Implementation of Page Replacement Techniques
   i. FIFO ii. LRU iii. Optimal page replacement

UNIT V	STORAGE MANAGEMENT	6+6

Mass Storage Structure: Overview of Mass Storage Structure- HDD scheduling – Swap Space Management, I/O systems: I/O Hardware, Application I/O interface, Kernel I/O Subsystem, File System Interface: File Concept – Access Methods – Directory Structure – Protection, File-System Implementation: File-System Structure- File-System Operations - Directory Implementation - Allocation Methods - Free-Space Management, - Case Study-Linux

List of Exercise/Experiments:

1. Simulation of File Allocation Techniques
   i. Sequential ii. Linked list iii. indexed
2. Implementation of File Organization Strategies
   Single level directory ii. Two level directory iii. Hierarchical level directory

TOTAL: 60 PERIODS

OUTCOMES:

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

CO1: Demonstrate the basic concepts of operating systems and process.
CO2: Implement process management techniques using inter-process communication.
CO3: Implement the concepts of process synchronization and deadlocks.
CO4: Apply various memory management schemes for the suitable scenario.
CO5: Describe various I/O and file management techniques.
CO6: Develop practical skills in developing system-level programming.

TEXTBOOKS:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts” II, 10th Edition, John Wiley and Sons Inc., 2018.
2. Andrew S Tanenbaum, "Modern Operating Systems", Pearson, 5th Edition, 2022 New Delhi.

REFERENCES:

1. William Stallings, "Operating Systems: Internals and Design Principles", 7th Edition, Prentice Hall, 2018.
2. Achyut S.Godbole, Atul Kahate, “Operating Systems”, McGraw Hill Education, 2016.

LIST OF EQUIPMENTS:

1. Standalone desktops with C/C++/Java/Equivalent compiler

24AM511	INTERNSHIP AND CAREER READINESS COURSE	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- To outline the basics of Data Warehouse concepts.
- To write queries using SQL and NoSQL.
- To discuss the features of python.
- To understand the fundamentals of Cloud.
- To familiarize the basic algorithms in AI, ML and Prompt Engineering.

MODULE I	Data Warehouse Concepts, SQL, NoSQL

Data Warehouse concepts: Need for BI, Data Warehouse, Key terminologies related to DWH architecture: OLTP vs OLAP, ETL, Data Mart, Metadata, DWH Architecture, creating a DWH

Data Lakehouse: Data Lake to Data Swamp, SQL Relational Databases, Transactional Processing, Relational Database Workload Types, Architectural Challenges, Databricks Evolution

ETL: Extract Data Dump from source, Data format consistency, Data Quality rules, Truncate & Load, Load strategies, Load Approach, Transform, Mapping, Enriching, Joins, filter, Remove Duplicates, Aggregation, Load, Dimension, Facts, EDW Tables, Data Marts

Variety of ETL Tools: Apache Airflow, Datastage, Oracle Data Integrator, SSIS, Talend, Hadoop, AWS Glue, Azure Data Factory, Google Cloud Dataflow, Stitch, SAP, Hevo, Qlik, Airbyte

Informatica: Informatica Architecture, Informatica PowerCenter & Repository, Informatica PowerCenter Designer, Informatica PowerCenter workflow manager, Informatica PowerCenter workflow monitor, Run Mappings, Workflow creation & Deletion

SQL (Beginner): DQL, DDL, DML, Filtering and sorting Data, Grouping and Aggregating Data, Joins and Subqueries, Window Functions, Optimizing SQL queries, Automation.

SQL (Advanced): Store Procedure, Trigger, Views, Functions.

NoSQL: NoSQL Fundamentals and Comparison with SQL

Power BI: Connecting Data Sources and Data Bases, Data Modeling, Creating Calculated Fields in Power BI

MODULE II	Python, Cloud Fundamentals

Python (Beginner): Variables, Operators, functions, Libraries, Methods, Refactoring, Enum, Tuples, Dictionaries, sets, Map, filter, reduce, Class & objects, Exceptions, Overloading

Python (Advanced): Iterators, Modules, Packages, Generators, List, Comprehensions, Regular expressions, Serialization, Partial functions, closures, Decorators

AWS: Benefits of AWS, AWS Services - Computer, Storage, Database Service, Networking Service, Security Service, Management tool Service, Developer tool Service

Azure: Cloud Computing, Services in Azure - Compute, Containers, Databases, Identity, Security, Networking, Storage

GCP: Cloud Computing, Benefits of GCP, GCP services, AWS vs Azure vs GCP

Python with Deep Learning: Python Data Science Libraries, Numpy, Scipy, Pandas, Matplotlib, Scikit-Learn, Statsmodels, Pandas, Sorting, Concatenate, Preprocessing - Time Series Data, Visualization

Python with AI: Introduction, Demand of AI, What is AI, Types of AI, Why python for AI, Python Packages for AI

MODULE III	AI, ML, Prompt Engineering

Artificial Intelligence: Artificial intelligence and its types, AI Roadmap, Machine learning and its types, Linear regression Analysis, Classifications in Machine Learning

Machine Learning: AI vs ML, Classification vs regression, Supervised learning, Unsupervised learning, Training Model, Preparing Data, K-Nearest Neighbors, Naive Bayes, Logistic Regression, Support Vector Machine, Neural Networks, Tensorflow, K-Means Clustering, Principal Component Analysis, K-Means and PCA Implementations

Prompt Engineering: Introduction to AI, Linguistics, Language Models, Prompt Engineering Mindset, Zero shot and few shot prompts, AI hallucinations, Vectors/text embeddings.

Generative AI Fundamentals: Generative AI and its use cases, How do LLMS (Large Language Models) work, LLMs generates output for NLP task, LLM model decision criteria, Proprietary models, Fine tuned models, Mixing LLM flavors in workflow, Data privacy, Data security

OUTCOMES:

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

CO1: Apply the basics of Data Warehouse concepts.
CO2: Apply ETL Tools for Data processing.
CO3: Write queries using SQL and NoSQL.
CO4: Apply the features of python.
CO5: Elaborate the fundamentals of Cloud and various services.
CO6: Demonstrate the basic algorithms in AI, ML and summarize the basics of Prompt Engineering.

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016.
2. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012
3. Jake Vander Plas, "Python Data Science Handbook: Essential Tools for Working with Data", Kindle Edition, 2017
4. Elmasri R. and S. Navathe, “Fundamentals of Database Systems”, Pearson Education, 7th Edition, 2016.
5. Brett Powell , Mastering Microsoft Power BI: Expert techniques for effective data analytics and business intelligence, Packt Publications, 2018.
6. Suresh Kumar Mukhiya and Usman Ahmed, “Hands-on Exploratory Data Analysis with Python”, Packt publishing, March 2020.
7. Ethem Alpaydin, “Introduction to Machine Learning, Adaptive Computation and Machine Learning Series”, Third Edition, MIT Press, 2015.
8. Nathan Hunter, The Art of Prompt Engineering with Chatgpt: A Hands-On Guide: 3 (Learn AI Tools the Fun Way!), Shroff/Hunter Publishers, 2023
9. Joseph Babcock and Raghav Bali, Generative AI with Python and TensorFlow 2, Packt Publications, 2021.

24GE511	PRODUCT DEVELOPMENT LAB – 3 (Simulation and Analysis Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Use the innovative design methodology to articulate the product concepts.
- Conduct basic tests to extract the qualitative and quantitative performance factors.
- Perform simulations, experiments, or case studies to test the reliability and relevance of the proposed design.

DESCRIPTION

The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. Further at the end of the semester, they will make a final presentation to exhibit the methodology and implementation of design process.

LIST OF ACTIVITIES:

1. Prepare the design concepts for the identified prototype.
2. Perform simulations, experiments, or case studies to test the reliability and relevance of the proposed design.
3. Present the methodology to be followed and the steps carried out in the implementation of design process.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Analyze critically the design process and outcomes to identify areas for improvement and future research directions.

CO2: Develop preliminary prototypes or models to demonstrate design concepts and evaluate their effectiveness.

CO3: Conduct iterative testing and evaluation of designs, incorporating feedback to improve outcomes.

CO4: Apply industry-standard tools, technologies, and frameworks to execute the design process effectively.

CO5: Explain design concepts through well-structured oral presentations, reports, and visual aids tailored to diverse audiences.

LIST OF EQUIPMENT

S. No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No

9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets
11	Open source simulation software	1

24HS511	PROFESSIONAL COMMUNICATION-I (TOEFL) 24HS511 LABORATORY COURSE (Common to all branches of Third Year (B.E./B.Tech)	L	T	P	C
		0	0	4	2

OBJECTIVES: The Course will enable learners to:

- Develop learners' ability to use TOEFL-specific reading strategies to comprehend academic texts and answer test-based questions.
- Enable learners to produce organized and grammatically accurate TOEFL-style writing for both integrated and independent tasks.
- Enhance learners' listening comprehension of academic lectures and conversations typical of the TOEFL iBT test.
- Build learners' speaking skills for TOEFL tasks, including clear organization of ideas, fluency, and intelligible pronunciation.
- Familiarize learners with integrated TOEFL tasks requiring the combined use of listening, reading, speaking, and writing under time constraints.

Unit – I	Introduction to TOEFL and LSRW	12

Purpose of the TOEFL - Types of TOEFL Exams - TOEFL Sections - Scoring and Grading -Test Administration -Test-Taking Tips-Overview of Listening, Speaking, Reading and Writing

UNIT II	Reading Skills – Part 1	12

TOEFL Reading Comprehension -Fast Reading Techniques and Skills-Increasing eye-span and other speed-reading techniques; Skimming, scanning, searching for gist; Understanding context and theme of passages - Understanding the author's tone- Reading Question Types - Question Types Identification Quiz - Detail Question Strategy Application- Purpose Question Strategy Application- Purpose Question Quiz - Inference Question Strategy Application-Inference Question Strategy Introduction- Inference Quiz Questions - Except/Not Question Strategy Introduction and Application - Except Question Quiz

UNIT – III	Writing Skills – Part 1	12

TOEFL Knowledge Guide and TOEFL Writing - Pitfalls of poor writing skills – Impact discussed in professional as well as recruitment contexts - Essentials of good writing skills - Clear ideation Simple sentences Coherent structure Correct punctuations - Ability to make a write-up understandable from a point of no context - Creative writing - Integrated Essay: Reading Note-Taking Introduction - Reading Note -Taking: Finding Author's POV Reading, Note-Taking: Finding Author's 3 Controlling Ideas - Listening Note-Taking: Academic Lecture Listening Note-Taking: Applying the Strategy - Integrated Essay: Writing Introductions Integrated Essay: Writing Body Paragraphs

UNIT IV	Listening Skills – Part 1	12

TOEFL Knowledge Guide and TOEFL Listening -Note Making Strategy – Introduction, General TOEFL Note-taking Strategies and Benefits- Top Ten Challenges Taking Lecture Notes, Rubric Analysis - Discourse Marker: Usage and Impact, Discourse Marker and Signposting; Listening Note Making Section (Conversations), Conversations Note-Making Strategies - Application of Note Making Strategies (Conversations) - Conversation Question Answering Strategy, Application of Answering Strategy (Conversations)

UNIT V	Speaking Skills – Part 1	12

TOEFL Speaking Course Introduction Speaking Course– Phonetics - Understanding right pronunciations and sounds - Word stress; Pitfalls of mispronounced words in professional contexts – case studies

TOTAL: 60 PERIODS

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

CO1: Apply TOEFL reading strategies (skimming, scanning, inference, vocabulary recognition) to answer academic reading questions with appropriate accuracy under test conditions.

CO2: Produce TOEFL-style integrated and independent essays that meet task requirements, demonstrating coherence, grammatical accuracy, and academic vocabulary within the allotted time.

CO3: Identify the main ideas, supporting details, and the speaker's attitudes in TOEFL-type academic listening passages, and respond correctly to comprehension questions.

CO4: Deliver organized spoken responses to TOEFL speaking prompts with intelligible pronunciation, adequate fluency, and relevant content within the prescribed time.

CO5: Complete TOEFL integrated tasks by synthesizing information from reading and listening inputs into coherent spoken or written responses using appropriate time-management strategies.

TEXT BOOKS:

1. Educational Testing Service. (2024). The Official Guide to the TOEFL® Test (7th ed.). Princeton,NJ:ETS. Educational Testing Service. (2024). Official TOEFL iBT® Tests, Volume 1 (5th ed.). Princeton,NJ:ETS.
2. Educational Testing Service. (2024). Official TOEFL iBT® Tests, Volume 2 (4th ed.). Princeton, NJ: ETS.

REFERENCE BOOKS & ONLINE RESOURCES BOOKS

1. Pamela J. Sharpe.Barron's TOEFL iBT. 17th ed., Barron's Educational Series, 2023.
2. Kaplan Test Prep.Kaplan's TOEFL iBT Prep Plus. 2024–2025 ed., Kaplan Publishing, 2024. 3. Educational Testing Service (ETS).The Official Guide to the TOEFL® Test. 7th ed., Princeton, NJ: ETS, 2024.
4. Educational Testing Service (ETS).Official TOEFL iBT® Tests, Volume 1. 5th ed., Princeton, NJ: ETS, 2024. 5. Educational Testing Service (ETS).Official TOEFL iBT® Tests, Volume 2. 4th ed., Princeton, NJ: ETS, 2024.

Online Resources

1.Coursera.TOEFL Preparation Specialization. Available at:https://www.coursera.org/specializations/toefl-preparation
2. Coursera.TOEFL Test-Taking Strategies. Available at:https://www.coursera.org/learn/toefl test-taking-strategies
3. Complete TOEFL Prep Course (Free Online). Available at:https://coursesity.com/course detail/complete-toefl-prep-course
4. Preply.TOEFL Preparation Course. Available at:https://preply.com/en/courses/english/toefl preparation-course

24CS511	ADVANCED APTITUDE AND CODING SKILLS - I	L	T	P	C
		0	0	3	1.5

OBJECTIVES:

- To develop vocabulary for effective communication and reading skills.
- To build the logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.

LIST OF EXERCISES:

1. English – Phase I Advanced

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase I Advanced

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase I Advanced

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase I

Logical, Compilation and Code reuse

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop vocabulary for effective communication and reading skills.

CO2: Build the logical reasoning and quantitative skills.

CO3: Develop error correction and debugging skills in programming.

SEMESTER VI

24AM603	AUTOMATA THEORY AND COMPILER DESIGN	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To introduce the fundamental concepts of automata theory.
- To understand deterministic and non-deterministic finite automata.
- To elaborate on Regular Expressions and Grammars.
- To introduce Push down Automata and Turing Machines.
- To introduce the major concepts of language translation and compiler design.
- To elaborate the code optimization and code generation in compiler design.

UNIT I	INTRODUCTION TO AUTOMATA THEORY	9

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.
Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.
Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with €-transitions to NFA without €-transitions. Conversion of NFA to DFA.

UNIT II	REGULAR EXPRESSIONS AND CONTEXT FREE GRAMMARS	9

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.
Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma.
Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

UNIT III	PDA AND TURING MACHINES	9

Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA and CFG's, Acceptance by final state
Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine .

UNIT IV	LEXICAL AND SYNTAX ANALYSIS	9

Introduction: The structure of a compiler,
Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex,
Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Parser Generators YACC.

UNIT V	CODE GENERATION AND OPTIMIZATION	9

Code generation and optimization: Issues in the design of code generator, a simple code generator, Introduction to code optimization, Basic blocks & flow graphs, DAG representation of basic blocks, Peephole optimization, the principle sources of optimization.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Construct deterministic and non-deterministic finite automata.
CO2: Design context free grammars for formal languages using regular expressions.
CO3: Use PDA and Turing Machines for recognizing context-free languages.
CO4: Design a lexical analyzer.
CO5: Design syntax analyzer.
CO6: Design a simple code generator and apply different code optimizations.

TEXT BOOKS:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory,

Languages, and Computation, 3rd Edition, Pearson Education, 2008.
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, "Compilers Principles, Techniques and Tools", Second Edition, Pearson, 2013.

REFERENCES:

1. K.L.P Mishra and Chandrashekaran, Theory of Computer Science – Automata languages and computation, 3rd Edition, PHI, 2007.
2. Elain Rich, "Automata, Computability and complexity", 1st Edition, Pearson Education, 2018.
3. Peter Linz, "An introduction to Formal Languages and Automata", Jones and Bartlett Publishers, 6th Edition, 2016.
4. K Muneeswaran, "Compiler Design", Oxford University Press, 2013.
5. John C Martin, Introduction to Languages and The Theory of Computation, TMH, 4th Edition, 2010.

24AM601	AGENTIC AI (Lab Integrated Course)	L	T	P	C
		3	0	2	4

OBJECTIVES:

The Course will enable learners to:

- Explain the architecture and design principles of autonomous AI agents
- Apply LLM prompting and tool-use patterns to build functional agentic systems
- Design and orchestrate multi-agent pipelines using modern frameworks
- Build and evaluate RAG-based knowledge-augmented agent systems
- Deploy agents responsibly with safety guardrails and regulatory awareness

UNIT I	FOUNDATIONS OF AGENTIC AI	9 + 6

From narrow AI to autonomous agents — historical trajectory and paradigm shift, Defining agentic systems: goals, perception, memory, action, and environment, The Sense–Plan–Act loop versus modern LLM-based ReAct patterns, Cognitive architectures: deliberative, reactive, and hybrid agents,

Agent types: goal-based, utility-based, model-based, and learning agents, Multi-agent systems: cooperation, competition, and emergent behavior, Ethical and safety considerations in autonomous decision-making.

Laboratory Programs:

1. Environment setup (Python / Node.js); implement a minimal goal-seeking agent on a grid-world simulation; compare greedy vs. BFS strategies
2. Implement a rule-based reactive agent and a deliberative agent side-by-side; log and analyse decision traces across multiple scenarios

UNIT II	LLMs AS BACKBONE AGENTS	9 + 6

Large language models: transformer architecture recap, tokenisation, and inference, Prompting strategies: zero-shot, few-shot, chain-of-thought, and self-consistency, ReAct (Reasoning + Acting): interleaving thought and tool calls, Tool use and function calling: schema design, structured outputs, and error recovery, Memory in LLM agents: in-context, episodic (vector stores), and parametric memory, Context-window management: chunking, summarisation, and sliding-window techniques, Hallucination, calibration, and grounding strategies for reliable agents

Laboratory Programs:

1. Prompt engineering: benchmark chain-of-thought versus direct prompting on reasoning tasks; measure and compare accuracy metrics
2. Implement a tool-calling agent using the Anthropic / OpenAI API; integrate web search and calculator tools; handle errors and edge cases

UNIT III	AGENT FRAMEWORKS AND ORCHESTRATION	9 + 6

Popular agent frameworks: LangGraph, AutoGen, CrewAI, and LlamaIndex Agents, Workflow design: sequential, parallel, conditional, and loop-based pipelines, Orchestrator–subagent patterns: task decomposition and delegation strategies, Model Context Protocol (MCP): standardising tool and resource exposure, Planning algorithms: hierarchical task networks, STRIPS, and LLM-driven planners, State management and checkpointing in long-horizon agentic tasks, Observability: tracing, logging, and evaluating pipelines using LangSmith and Arize.

Laboratory Programs:

1. Build a multi-step research agent with LangGraph; implement conditional branching based on LLM output confidence and intermediate results.
2. Configure a custom MCP server exposing a database tool; connect to an orchestrator and trace the full execution flow end-to-end.

UNIT IV	RAG AND KNOWLEDGE AGENTS	9 + 6

RAG fundamentals: document ingestion, chunking strategies, and embedding models, Vector databases: FAISS, Pinecone, Chroma, Weaviate — indexing and ANN search, Advanced RAG: hybrid search, re-ranking, HyDE, and multi-hop retrieval, Graph RAG: knowledge graphs as structured memory for agents, Agentic RAG: query rewriting, self-correction loops, and citation grounding, Fine-tuning versus RAG: when to adapt model weights versus retrieval augmentation, Evaluation metrics: RAGAS, faithfulness, answer relevance, context precision and recall.

Laboratory Programs:

1. Build a full RAG pipeline over a custom PDF corpus using LlamaIndex and ChromaDB; benchmark retrieval quality with RAGAS metrics
2. Implement an agentic RAG system with self-correction: agent re-queries when retrieved context is insufficient; visualise retrieval quality improvements

UNIT V	DEPLOYMENT, SAFETY AND APPLICATIONS	9 + 6

Production deployment: containerisation (Docker), API serving (FastAPI), and cloud hosting, Agent safety: sandboxing, tool permission scoping, and human-in-the-loop mechanisms, Guardrails: input/output filtering, constitutional AI, and red-teaming agents, Prompt injection attacks and defence strategies for agentic systems, Multi-modal agents: vision, audio, and code-execution capabilities, Domain applications: coding agents, customer service bots, scientific research agents, and robotic process automation, Regulatory landscape: EU AI Act, NIST AI RMF, and responsible deployment practices

Laboratory Programs:

1. Red-team a peer's agent: craft adversarial prompts, document vulnerabilities, produce a structured security report, and propose fixes
2. Mini-project: design, build, and demo a domain-specific agentic application with full deployment pipeline (Docker + FastAPI); present and defend design decisions.

TOTAL: 45 + 30 = 75 PERIODS

OUTCOMES:

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

CO1: Explain the architecture and design principles of autonomous AI agents

CO2: Apply LLM prompting and tool-use patterns to build functional agentic systems

CO3: Design and orchestrate multi-agent pipelines using modern frameworks

CO4: Build and evaluate RAG-based knowledge-augmented agent systems

CO5: Deploy agents responsibly with safety guardrails and regulatory awareness

TEXT BOOKS:

1. Russell, S. & Norvig, P. — Artificial Intelligence: A Modern Approach, 4th ed. Pearson, 2020.
2. Wooldridge, M. — An Introduction to MultiAgent Systems, 2nd ed. Wiley, 2009.

REFERENCES:

1. Yao et al. — ReAct: Synergizing Reasoning and Acting in Language Models. ICLR 2023.
2. Gao et al. — Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS 2020. (arXiv 2005.11401)
3. Wang et al. — A Survey on Large Language Model-based Autonomous Agents. (arXiv 2308.11432)
4. Wei et al. — Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS 2022.

24AM602	Data Visualization	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To outline exploratory data analysis and the phases involved in data analysis.
- To discuss various statistical techniques for data analysis.
- To demonstrate the basics of feature engineering on different types of data.
- To perform data analysis and apply visualization techniques.
- To apply the methods of time series analysis.
- To formulate dashboards using different datasets by applying data engineering and feature extraction techniques.

UNIT I	EXPLORATORY DATA ANALYSIS	6+6

EDA fundamentals – Understanding data science – Significance of EDA – Making sense of data – Comparing EDA with classical and Bayesian analysis – Software tools for EDA.

Visual Aids For EDA- Data transformation techniques-merging database, reshaping and pivoting, Transformation techniques - Descriptive Statistics-types of kurtosis, quartiles, Grouping Datasets-data aggregation, group wise transformation.

List of Exercise/Experiments

1. Install the following Data Mining and data Analysis tool: Weka, KNIME, Tableau Public.
2. Perform exploratory data analysis (EDA) on with datasets like email data set. Export all your emails as a dataset, import them inside a pandas data frame, visualize them and get different insights from the data.

UNIT II	FEATURE ENGINEERING	6+6

Text Data – Visual Data – Feature-based Time-Series Analysis – Data Streams – Feature Selection and Evaluation.

List of Exercise/Experiments

1. Implement document embeddings for fake news identification.
2. Implement feature based representations of time series
3. Implement feature selection algorithm for data streams

UNIT III	VISUALIZING DATA	6+6

The Seven Stages of Visualizing Data, Processing-load and displaying data – functions, sketching and scripting, Mapping - Location, Data, two sided data ranges, smooth interpolation of values over time - Visualization of numeric data and non-numeric data.

List of Exercise/Experiments

1. Perform text mining on a set of documents and visualize the most important words in a visualization such as word cloud.
2. Perform Data Analysis and representation on a Map using various Map data sets with Mouse Rollover effect, user interaction, etc..
3. Build cartographic visualization for multiple datasets involving various countries of the world; states and districts in India etc.

UNIT IV	TIME SERIES ANALYSIS	6+6

Overview of time series analysis - showing data as an area, drawing tabs, handling mouse input, Connections and Correlations – Preprocessing-introducing regular expression, sophisticated sorting, Scatterplot Maps-deployment issues.

List of Exercise/Experiments

1. Perform Time Series Analysis with datasets like Open Power System Data.
2. Build a time-series model on a given dataset and evaluate its accuracy.

UNIT V	TREES, HIERARCHIES, AND RECURSION	6+6

Treemaps - treemap library, directory structure, maintaining context, file item, folder item, Networks and Graphs-approaching network problems-advanced graph example, Acquiring data, Parsing data.

List of Exercise/Experiments

1. Use a case study on a data set and apply the various visualization techniques and present an analysis report.
2. Mini-Project:- Create a Dashboard for a dataset with a visualization tool.

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Outline exploratory data analysis and the phases involved in data analysis.
CO2: Demonstrate various statistical techniques for data analysis.
CO3: Present the basics of feature engineering on different types of data.
CO4: Perform data analysis and apply visualization techniques.
CO5: Apply the methods of time series analysis.
CO6: Develop dashboards using different datasets by applying data engineering and feature extraction techniques.

TEXT BOOKS:

1. Suresh Kumar Mukhiya and Usman Ahmed, “Hands-on Exploratory Data Analysis with Python”, Packt Publishing , First Edition, March 2020.
2. Guozhu Dong, Huan Liu, "Feature Engineering for Machine Learning and Data Analytics", First Publication, CRC Press, First edition, 2018.
3. Ben Fry, “Visualizing Data”, O’reilly Publications, First Edition, 2007.

REFERENCES:

1. Danyel Fisher & Miriah Meyer, “Making Data Visual: A Practical Guide To Using Visualization For Insight”, O’reilly publications, 2018.
2. Claus O. Wilke, ”Fundamentals of Data Visualization”, O’reilly publications, 2019.
3. EMC Education Services, “Data Science and Big data analytics: Discovering, Analyzing, Visualizing and Presenting Data”, Wiley Publishers, 2015.
4. Tamara Munzner, “Visualization Analysis and Design”, A K Peters/CRC Press; 1st edition, 2014.
5. Matthew O. Ward, Georges Grinstein, Daniel Keim, “Interactive Data Visualization: Foundations, Techniques, and Applications”, 2nd Edition, CRC press, 2015.

LIST OF EQUIPMENTS:

1. Systems with Python/R, Tableau Public / PowerBI

24GE611	PRODUCT DEVELOPMENT LAB – 4 (Prototype and Validation Phase) (Common to All Branches)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

- Analyze the real time problems during project/product development in engineering perspective.
- Develop comprehensive report on the engineering facts applied to a specific problem.
- Evaluate the effectiveness of the product or a system through the knowledge acquired.
- Synthesize the business opportunities for a new product with novel design.

DESCRIPTION

The students may be grouped into a batch of strength 3 or 4 to work under a project supervisor. The student batches should present their work done, process involved at the end of the semester and submit a report.

LIST OF ACTIVITIES:

1. Develop a prototype based on the proposed design.
2. Demonstration of the project/product and submission of report.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Understand and explain the real time problems.
CO2: Analyze the methods to develop solution to the systems.
CO3: Evaluate the performance and effectiveness of the existing systems.
CO4: Summarize and prepare reports for the experimental determinations.
CO5: Classify, compare and analyze business opportunities for a new product.

LIST OF EQUIPMENT

S. No.	Description of Equipment	Quantity
1	CNC Router	1 No
2	3D Printer	1 No
3	3D Scanner	1 No
4	Laser cutting Machine	1 No
5	Multimeter	5 Nos
6	Solder Stations	5 Sets
7	Desoldering Machine	1 No
8	PCB Milling Machine	1 No
9	Variable Power Supply	1 No
10	Electronic Components like Resistors, Transistors, Diode, Inductor, Capacitor, etc.	5 Sets

24HS611	PROFESSIONAL COMMUNICATION- II (TOEFL)	L	T	P	C
		0	0	2	1

OBJECTIVES:

The Course will enable learners to:

● Develop effective reading strategies, with a primary focus on TOEFL-oriented tasks, to enhance comprehension.

● Strengthen note-taking skills and organize written responses coherently with TOEFL- related writing tasks.

● Enhance active listening skills to effectively comprehend academic lectures and campus conversations, in alignment with TOEFL iBT.

● Demonstrate confidence, fluency, and clarity while presenting ideas, in alignment with TOEFL integrated speaking tasks.

● Apply Listening, Speaking, Reading, and Writing (LSRW) skills effectively to achieve overall communicative competence, with TOEFL- integrated task requirements.

UNIT - I	Reading Skills – Part 2	6

Paraphrase Question Strategy Application, Paraphrase Question Strategy and Tactics Process, Question Quiz-Sentence, Addition Question Type Strategy, Sentence Addition Quiz-Vocabulary Question Strategy Application, Vocabulary Through Context-Reference Question Strategy Intro and Application, Reference, Vocabulary, How Organized Strategy-Non-Category Questions Strategy, Summary and Category (table) Question Strategy, Summary Question Answer Explanation-TOEFL Reading Comprehension Rubrics Analysis.

UNIT - II	Writing Skills – Part 2	6

Listening Note-Taking: Academic Lecture Listening Note-Taking: Applying the Strategy-Integrated Essay: Writing Introductions Integrated Essay: Writing Body Paragraphs-Writing Thesis Statements Writing Introductions: Writing Topic Sentences: Topic Sentences-Writing Body Paragraphs-Eliminate Weak Sentence-Full Essay writing: Agree – Disagree, Support – Oppose, If - Imaginary Questions, Compare and Contrast, Advantage and Disadvantage , Standard Opinion Question , Discuss and Choose Side to Agree -Rubric Analysis.

UNIT - III	Listening Skills – Part 2	6

Application of Note-Making Strategy 2 (Conversations)-Application of Answering Strategy 2 (Conversations)-Application of Note-Making Strategy 3 (Conversations)-Application of Answering Strategy 3 (Conversations)-Application of Note-Making Strategy 4 (Conversations)-Application of Answering Strategy 4 (Conversations)-Demonstration (Questions Go In-Order) Timed Strategy Application Quiz.

UNIT – IV	Speaking Skills – Part 2	6

Personal Opinion Questions- Introduction, Note-Making: Personal Opinion Questions- Speaking Template, Useful Expressions, In-Class Practice-Campus Announcement Introduction Overview: Question Reading, Text Note-Taking, Teacher Sample Responses-Student Practice: Campus Announcement Question ,Speech-to-Text Teacher Demonstration Question, Speech-to-Text Student Opportunity-Academic Text-Lecture Question: Introduction Overview, Reading Text Note-Taking, Lecture- Listening Note-Making Academic Text Lecture, Teacher Sample Response and Student Practice Opportunity, Transcription of Speaking Responses: Academic Text - Lecture Question, Speech-to-Text Teacher Demonstration, Speech-to-Text Student Opportunity.

UNIT - V	Integrated LSRW approach	6

Listening: Active listening strategies such as note-taking and paraphrasing-Listening to various audio materials to improve comprehension and retention-Focuses on identifying main ideas, supporting details, and speaker attitudes. Speaking: Discussion on strategies for improving pronunciation, intonation, and fluency-Speaking on different topics to enhance verbal communication skills-Reading: Review the strategies for active reading, including skimming, scanning, and annotating texts-Practice reading across various genres to improve comprehension and vocabulary acquisition-Emphasize understanding the author's purpose, tone, and argumentation in texts-Writing: Discussion on principles of effective writing, including organization, coherence, and clarity- Practice writing essays or responses to prompts to develop written communication skills- Feedback on grammar, punctuation, and sentence

structure to improve writing proficiency.

TOTAL: 30 PERIODS

OUTCOMES:

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

CO1: Apply effective reading strategies to read academic texts fluently and respond accurately to TOEFL-style contextual questions.

CO2: Compose written responses with coherence, clarity, and logical flow in alignment with TOEFL integrated writing tasks.

CO3: Demonstrate active listening skills by comprehending academic lectures and campus conversations as required in the TOEFL Listening section, and respond appropriately.

CO4: Engage confidently in spoken communication by effectively performing TOEFL-integrated Speaking tasks with clarity and fluency.

CO5: Execute Listening, Speaking, Reading, and Writing (LSRW) skills, strategically to meet the standards of TOEFL iBT integrated tasks..

TEXT BOOKS:

1. Gear, J., & Gear, R. (2014). Cambridge preparation for the TOEFL test (4th ed.). Cambridge University Press.

2. Sharpe, P. J. (2020). Barron's TOEFL iBT (17th ed.). Barron's Educational Series

Reference Books & Online Resources Books

1. Educational Testing Service. (2023). The official guide to the TOEFL iBT test. McGraw-Hill Education.

2. Brown, H. D., & Lee, H. (2015). Teaching by principles: An interactive approach to language pedagogy (4th ed.). Pearson Education.

3. Richards, J. C. (2008). Teaching listening and speaking: From theory to practice. Cambridge University Press.

4. Anderson, N. J. (2008). Practical English language teaching: Reading. McGraw-Hill. 5. Hedge, T. (2005). Writing (2nd ed.). Oxford University Press.

Online Resources

1. Educational Testing Service. (n.d.). TOEFL iBT test content and structure. ETS. https://www.ets.org/india/toefl/test-takers/ibt/about/content.html

2. Exam Word. (n.d.). TOEFL test preparation resources. https://www.examword.com/d/toefl 3. Coursera. (n.d.). TOEFL reading and listening sections skills mastery. https://www.coursera.org/learn/toefl-reading-listening-sections-skills-mastery

24CS611	ADVANCED APTITUDE AND CODING SKILLS - II	L	T	P	C
		0	0	3	1.5

OBJECTIVES:

- To develop advanced vocabulary for effective communication and reading skills.
- To build an enhanced level of logical reasoning and quantitative skills.
- To develop error correction and debugging skills in programming.
- To apply data structures and algorithms in problem solving.

LIST OF EXERCISES:

1.English – Phase II Advanced

Vocabulary: Synonyms, Antonyms, Grammar: Subject-Verb Agreement, Tenses and Articles, Prepositions and Conjunctions, Speech and Voices, Comprehension: Inferential and Literal Comprehension, Contextual Vocabulary, Comprehension ordering

2. Logical Reasoning – Phase II Advanced

Deductive Reasoning: Coding deductive logic, Directional sense, Blood relations, Objective Reasoning, Selection decision tables, Puzzles, Inductive reasoning: Coding pattern and Number series pattern recognition, Analogy and Classification pattern recognition, Abductive Reasoning: Logical word sequence, Data sufficiency

3. Quantitative Ability - Phase II Advanced

Basic Mathematics: Divisibility, HCF and LCM, Numbers, decimal fractions and power, Applied Mathematics: Profit and Loss, Simple and Compound Interest, Time, Speed and Distance, Engineering Mathematics: Logarithms, Permutation and Combinations, Probability

4. Automata Fix – Phase II

Logical, Compilation and Code reuse

5. Automata - Phase II

Data Structure Concepts: Array and Matrices, Linked list, String processing and manipulation, Stack/Queue, Sorting and Searching Advanced Design and Analysis Techniques: Greedy Algorithms, Minimum Spanning Trees, String Matching, Divide and Conquer, Computational Geometry

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop advanced vocabulary for effective communication and reading skills.

CO2: Build an enhanced level of logical reasoning and quantitative skills.

CO3: Develop error correction and debugging skills in programming.

CO4: Apply data structures and algorithms in problem solving.

SEMESTER 7

24AM701	COMPUTER VISION	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamental concepts related to Image formation and processing.
- To learn feature detection, matching and detection.
- To become familiar with feature based alignment and motion estimation.
- To develop skills on 3D reconstruction.
- To understand image based rendering and recognition.

UNIT I	INTRODUCTION TO IMAGE FORMATION AND PROCESSING	9

Computer Vision - Geometric primitives and transformations - Photometric image formation - The digital camera - Point operators - Linear filtering - More neighborhood operators - Fourier transforms - Pyramids and wavelets - Geometric transformations - Global optimization.

UNIT II	FEATURE DETECTION, MATCHING AND SEGMENTATION	9

Points and patches - Edges - Lines - Segmentation - Active contours - Split and merge - Mean shift and mode finding - Normalized cuts - Graph cuts and energy-based methods.

UNIT III	FEATURE-BASED ALIGNMENT & MOTION ESTIMATION	9

2D and 3D feature-based alignment - Pose estimation - Geometric intrinsic calibration - Triangulation - Two-frame structure from motion - Factorization - Bundle adjustment - Constrained structure and motion - Translational alignment - Parametric motion - Spline-based motion - Optical flow - Layered motion.

UNIT IV	3D RECONSTRUCTION	9

Shape from X - Active range finding - Surface representations - Point-based representations- Volumetric representations - Model-based reconstruction - Recovering texture maps and albedosos

UNIT V	IMAGE-BASED RENDERING AND RECOGNITION	9

View interpolation Layered depth images - Light fields and Lumi graphs - Environment mattes - Video-based rendering-Object detection - Face recognition - Instance recognition - Category recognition - Context and scene understanding- Recognition databases and test sets.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Analyze and apply basic image processing techniques in practical applications.
CO2: Compare the concepts related to feature detection, matching and detection.
CO3: Implement feature-based alignment and motion estimation in real-world applications.
CO4: Create and Apply 3D Reconstruction techniques in diverse applications.
CO5: Perform image-based rendering and recognition.
CO6: Implement efficient solutions to image processing and computer vision problems.

TEXT BOOKS:

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London

Limited 2011.
2. Computer Vision: A Modern Approach, D. A. Forsyth, J. Ponce, Person Education, Second Edition, 2015

REFERENCES:

1. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004.
2. Christopher M. Bishop; Pattern Recognition and Machine Learning, Springer, 2006
3. E. R. Davies, Computer and Machine Vision, Fourth Edition, Academic Press, 2012.

24AM702	REINFORCEMENT LEARNING TECHNIQUES	L	T	P	C
		2	0	2	3

OBJECTIVES:

- To outline the concepts of Reinforcement Learning
- To solve problems using Monte Carlo Decision Process and Dynamic Programming.
- To implement problems using temporal difference learning.
- To apply functional approximation in reinforcement learning.
- To implement Deep Reinforcement Learning.
- To solve real-world problems using Reinforcement Learning.

UNIT I	INTRODUCTION TO REINFORCEMENT LEARNING	6+6

Introduction - Elements of RL, History of RL- Limitation and Scope - Examples – Multi-arm Bandits – k-armed Bandit Problem – Action-Value Methods – Incremental Implementation – Nonstationary Problem – Optimistic Initial Values – Upper Confidence Bound Action Selection – Gradient Bandit Algorithms – Contextual Bandits.

Lab Exercise:

1. **Implement the k-Armed Bandit problem** using Action-Value methods and compare the performance of different action selection strategies.
2. **Implement and compare Upper Confidence Bound (UCB) Action Selection and Gradient Bandit Algorithms** for solving the Multi-Armed Bandit problem.

UNIT II	MONTE DECISION PROCESS AND DYNAMIC PROGRAMMING	6+6

Finite Markov Decision Processes - The Agent Environment interface - Goals and Rewards – Returns and Episodes – Episodic and Continuing Tasks – Policies and Value Functions – Optimality and Approximation - Dynamic Programming - Policy Evaluation - Policy Improvement, Iteration - Value Iteration - Asynchronous DP- Efficiency of DP - Monte Carlo Prediction - Monte Carlo Estimation of Action Values - Monte Carlo Control - Off - policy Monte Carlo Prediction – Off-Policy Monte Carlo Controls.

Lab Exercise:

1. **Implement Policy Iteration** to determine the optimal policy and value function for a finite Markov Decision Process (MDP).
2. **Implement Monte Carlo Prediction and Monte Carlo Control** to estimate state-value and action-value functions for an episodic reinforcement learning environment.

UNIT III	TEMPORAL DIFFERENCE LEARNING	6+6

Temporal-Difference prediction – Advantages of TD Prediction Methods -Optimality of TD(0) – Sarsa: On-policy TD Control – Q Learning: Off-Policy TD Control – Expected Sarsa - n-step TD Prediction – n-step Sarsa - n-step Off-Policy Learning.

Lab Exercise:

1. **Implement the SARSA algorithm** to learn the optimal policy in a reinforcement learning environment.
2. **Implement Q-Learning and Expected SARSA** and compare their learning performance.

UNIT IV	FUNCTION APPROXIMATION METHODS	6+6

On-Policy Prediction with Approximation: Value-function Approximation – The Prediction Objective – Stochastic-gradient and Semi-gradient Methods - Linear Methods – Feature Construction for Linear Methods - Eligibility Traces: The λ-return – TD(λ).

Lab Exercise:

1. **Implement Linear Value Function Approximation** using stochastic gradient methods for policy prediction.
2. **Implement TD(λ) with Eligibility Traces** and analyze the effect of different λ values on learning performance.

UNIT V	DEEP REINFORCEMENT LEARNING	6+6

Deep Q-Learning – Deep Q-Networks - Rainbow DQN – Policy Gradient Methods – Policy Approximation and its advantages – Policy Gradient Theorem – REINFORCE: Monte Carlo Policy Gradient - Actor-Critic Methods - Hierarchical RL – Multi-Agent RL.

Lab Exercise:

1. **Implement a Deep Q-Network (DQN)** to solve a reinforcement learning environment and evaluate its performance.
2. **Implement an Actor-Critic method** for policy optimization and compare its performance with Deep Q-Learning.

TOTAL: 30 + 30=60 PERIODS

OUTCOMES:

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

CO1: Outline the concepts of Reinforcement Learning.
CO2: Solve problems using Dynamic Programming and Monte Carlo Decision Process.
CO3: Implement the concept of Temporal difference Learning (TML) to solve real world problems.
CO4: Apply functional approximation in reinforcement learning.
CO5: Implement Deep Reinforcement Learning to solve real world problems.
CO6: Solve real-world problems using Reinforcement Learning

TEXT BOOKS:

1. Sutton R. S. and Barto A. G., "Reinforcement Learning: An Introduction", MIT Press, Second Edition, 2020.
2. Phil Winder, “Reinforcement Learning: Industrial Applications of Intelligent Agents“. Oreilly, 2021.

REFERENCES:

1.Nimish Sanghi, “Deep Reinforcement Learning with Python with PyTorch, TensorFlow and OpenAI Gym”, Apress, First Edition, 2021.
2.Maxim Lapan, “Deep Reinforcement Learning Hands-On”, Packt, Second Edition, 2020.
3.Miguel Morales, “Deep Reinforcement Learning “, Miguel Morales, Manning Publication, 2020.
4.Kevin Murphy, “Machine Learning - A Probabilistic Perspective” , MIT press, 2012.
5.Christopher Bishop, “Pattern Recognition and Machine Learning”, Springer, 2006.

PROFESSIONAL ELECTIVES
DATA SCIENCE ANALYTICS

24AM901	FOUNDATIONS OF DATA SCIENCE	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To learn the fundamentals of Data Science.
- To describe and understand data for analysis.
- To describe and analyze relationships between variables.
- To apply various data pre-processing strategies.
- To apply data wrangling on data for further processing.

UNIT I	INTRODUCTION	6+6

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – data preparation - Exploratory Data analysis – build the model – presenting findings and building applications - Data Mining - Data Warehousing – Basic statistical descriptions of Data.

List of Exercise/Experiments:

1. Explore a dataset and describe it using basic statistics and represent it using simple graphs.
2. Group data and find averages or counts for each group.
3. Build a simple machine learning model from a dataset and show how well it works.

UNIT II	DESCRIBING DATA	6+6

Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores.

List of Exercise/Experiments:

1. Use graphs to find patterns or trends in a dataset.
2. Compare two or more columns to find relationships in a dataset.
3. Find the average, spread, and z-scores for numbers in a dataset.
4. Find which values are close to the average and which are far using z-scores.

UNIT III	DESCRIBING RELATIONSHIPS	6+6

Correlation –Scatter plots –correlation coefficient for quantitative data –computational formula for correlation coefficient –

Regression –regression line –least squares regression line –

Standard error of estimate – interpretation of r2 –multiple regression equations –regression towards the mean.

List of Exercise/Experiments:

1. Use a scatter plot to see how two variables are related. Calculate the correlation coefficient to measure the strength of the relationship.
2. Find the line of best fit (regression line) for two variables. Calculate r^2 to see how well the line fits the data.
3. Use a Multiple Linear Regression model to predict a target variable from multiple inputs, and plot the predicted vs. actual values to evaluate model performance.
4. Create a dashboard or summary using charts and key numbers.

UNIT IV	DATA PREPROCESSING	6+6

Data Preprocessing – Purpose – Tools – Levels – Cleaning up the Table – Unpacking – Restructuring – Reformulating the Table - Data Cleaning: – Missing Values – Outliers – Errors - Data Fusion vs Data Integration – Directions – Adding attributes – data objects - Entity identification – Data Reduction vs data redundancy – Types – Performing Numerosity Data reduction – Sampling – Principal Component Analysis – Data Transformation – Normalization and Standardization – Discretization – Smoothing – Aggregation - Binning.

List of Exercise/Experiments:

1. Clean and preprocess a dataset by handling missing values, scaling features, treating outliers, and creating dummy variables.
2. Train and tune a model using cross-validation to understand and manage the bias-variance trade-off.
3. Perform Outlier Analysis on any data set.

UNIT V	DATA WRANGLING	6+6

Processing Uni-dimensional Data: Creating Vectors - Inspecting the Data Distribution with Histograms- Aggregating Numerical Data - Arithmetic Operators - Indexing vectors- Multidimensional data: Creating matrices - Reshaping - visualization (2d,3d) - Heterogeneous Data: Data frames - Aggregation - Transformation - Indexing - Accessing Database: Filtering - Ordering - Removing Duplicates - Grouping and Aggregating - Joining - Handling with Many Files.

List of Exercise/Experiments:

1. Create and reshape matrices, apply matrix operations, and visualize data using 2D heatmaps and 3D surface plots.
2. Build a DataFrame, perform aggregation and transformation, filter and sort data, remove duplicates, group, and join tables.
3. Simulate SQL-style operations like filtering, ordering, deduplication, and grouping in a DataFrame.

TOTAL:30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Explain the fundamentals of data science.
CO2: Illustrate the basics of data for analysis.
CO3: Identify relationships between variables.
CO4: Implement various data pre-processing strategies.
CO5: Implement data wrangling for further processing of data.
CO6: Apply Data preprocessing on real-time data sets.

TEXT BOOKS:

1. Robert S. Witte and John S. Witte, “Statistics”, Eleventh Edition, Wiley Publications, 2017. (Units 2, 3)
2. Jacqueline Kazil and Katharine Jarmul, Data Wrangling with Python: Tips and Tools to Make Your Life Easier, O'Reilly Media, 1st Edition, 2016.

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016. (Unit 1)
2. Roy Jafari, Hands-On Data Preprocessing in Python: Learn how to effectively prepare data for successful data analytics, Packt Publications, First Edition, 2022. (Unit 4)
3. Marek Gagolewski, Minimalist Data Wrangling with Python, Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0), 2022. (Unit 5)
4. Hui Lin, Ming Li, Practitioner’s Guide to Data Science, 1st Edition, Chapman and Hall/CRC, 2023.
5. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012.
6. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data, Packt Publications, 2020.
7. Dr. Tirthajyoti Sarkar, Shubhadeep Roychowdhury, Data Wrangling with Python: Creating actionable data from raw sources, Packt Publications, First Edition, 2019.
8. Robert H. Shumway, David S. Stoffer, Time Series Analysis and Its Applications With R Examples, Fourth Edition, 2016, Springer.
9. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements Of Statistical Learning: Data Mining, Inference, and Prediction, Springer, Second Edition, 2017.

10. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, Third Edition,2022.
11. Jake VanderPlas, “Python Data Science Handbook – Essential tools for working with data”, O’Reilly, 2017.
12. Ashwin Pajankar, Aditya Joshi, Hands-on Machine Learning with Python: Implement Neural Network Solutions with Scikit-learn and PyTorch, Apress, 2022.
13. NPTEL Courses:
    a. Data Science for Engineers - https://onlinecourses.nptel.ac.in/noc23_cs17/preview
    b. Python for Data Science - https://onlinecourses.nptel.ac.in/noc23_cs21/preview

LIST OF EQUIPMENTS:

Systems with Anaconda, Jupyter Notebook, Python, Pandas, NumPy, MathPlotlib

24AM902	TEXT AND SPEECH ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To introduce the tools and techniques for performing text and speech analytics in diverse contexts.
- To understand the tools and technologies involved in developing text and speech applications.
- To demonstrate the use of computing for building applications in text and speech processing.
- To use information Retrieval Techniques to build and evaluate text processing systems.
- To apply advanced speech recognition methodologies in practical applications.

UNIT I	TEXT PROCESSING	9

Speech and Language Processing - Regular Expression - Text normalization – Edit Distance - Lemmatization – Stemming – N-gram Language Models - Vector Semantics and Embeddings.

UNIT II	TEXT CLASSIFICATION	9

Text Classification Tasks – Language Model – Neural Language Models – RNNs as Language Models – Transformers and Large Language Models.

UNIT III	QUESTION ANSWERING AND DIALOGUE SYSTEMS	9

Information Retrieval – Dense Vectors – Neural IR for Question Answering – Evaluating Retrieval-based Question Answering – Frame-based Dialogue Systems – Dialogue Acts and Dialogue State – Chatbots – Dialogue System Design.

UNIT IV	TEXT TO SPEECH SYNTHESIS	9

Automatic Speech Recognition Task – Feature Extraction for ASR: Log Mel Spectrum – Speech Recognition Architecture – CTC - ASR Evaluation: Word Error Rate – TTS – Speech Tasks.

UNIT V	SPEECH RECOGNITION	9

LPC for speech recognition - Hidden Markov Model (HMM) - Training procedure for HMM- subword unit model based on HMM - Language models for large vocabulary speech recognition - Overall recognition system based on subword units - Context dependent subword units- Semantic post processor for speech recognition.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Apply the fundamental techniques in text processing for various NLP tasks.
CO2: Implement advanced language models and improve text classification accuracy.
CO3: Designing text processing systems using state-of-the-art techniques.
CO4: Design, implement, and evaluate ASR and TTS systems.
CO5: Apply advanced speech recognition methodologies in practical applications.
CO6: Use information Retrieval Techniques to build and evaluate text processing systems.

TEXT BOOKS:

1. Jurafsky, D. and J. H. Martin, Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Pearson Publication, Third Edition, 2022.
2. Lawrence Rabiner, Biing-Hwang Juang and B.Yegnanarayana, “Fundamentals of Speech Recognition”, Pearson Education, 2009.

REFERENCES:

1. John Atkinson-Abutridy, Text Analytics: An Introduction to the Science and Applications of Unstructured Information Analysis, CRC Press, 2022.
2. Jim Schwoebel, NeuroLex, Introduction to Voice Computing in Python, 2018
3. Lawrence R. Rabiner, Ronald W. Schafe, Theory and Applications of Digital Speech Processing, First Edition, Pearson, 2010.
4. Srinivasa-Desikan, Bhargav. Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras. Packt Publishing Ltd, 2018.

24AM903	NoSQL DATABASES	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

1. Understand the fundamental principles of NoSQL databases, including aggregate data models, distribution models, and the need for polyglot persistence.
2. Analyze consistency, version stamps, and map-reduce patterns in the context of distributed NoSQL systems.
3. Design and implement data models using key-value, document, column-family, and graph databases.
4. Perform CRUD operations, indexing, and query optimization across different NoSQL database types.
5. Apply schema migration techniques and polyglot persistence strategies for real-world applications.
6. Evaluate and choose appropriate NoSQL databases for specific application requirements.

UNIT I	WHY NoSQL AND AGGREGATE DATA MODELS	6+6

Why NoSQL? – The Value of Relational Databases – Getting at Persistent Data – Concurrency – Integration – Impedance Mismatch – Application and Integration Databases – Attack of the Clusters – The Emergence of NoSQL – Aggregate Data Models – Aggregates – Example of Relations and Aggregates – Consequences of Aggregate Orientation – Key-Value and Document Data Models – Column-Family Stores.

List of Exercises:

1. Install and configure MongoDB, Redis, and Neo4j; verify connectivity and perform basic shell commands.
2. Create a relational schema in MySQL and an equivalent aggregate data model in MongoDB for a simple e-commerce product catalog; compare the structures.
3. Implement basic key-value pair storage and retrieval operations using Redis.

UNIT II	DATA MODELS, DISTRIBUTION, AND CONSISTENCY	6+6

More Details on Data Models – Relationships – Graph Databases – Schemaless Databases – Materialized Views – Distribution Models – Single Server – Sharding – Master-Slave Replication – Peer-to-Peer Replication – Combining Sharding and Replication – Consistency – Update Consistency – Read Consistency – Relaxing Consistency – The CAP Theorem – Relaxing Durability – Quorums.

List of Exercises:

1. Create and query a graph database in Neo4j; model a social network with nodes and relationships.
2. Configure a MongoDB replica set to demonstrate master-slave replication and test failover behavior.
3. Demonstrate the CAP theorem trade-offs by simulating network partition scenarios using a distributed NoSQL setup.

UNIT III	VERSION STAMPS, MAP-REDUCE, AND NoSQL DATABASE TYPES	6+6

Version Stamps – Map-Reduce – Key-Value Databases – What Is a Key-Value Store – Key-Value Store Features – Consistency – Transactions – Query Features – Structure of Data – Scaling – Suitable Use Cases – Storing Session Information – User Profiles, Preferences – Shopping Cart Data – Document Databases – What Is a Document Database – Features – Consistency – Transactions – Availability – Query Features – Scaling – Suitable Use Cases – Event Logging – Content Management Systems, Blogging Platforms – Web Analytics or Real-Time Analytics – E-Commerce Applications.

List of Exercises:

1. Implement a session management system using Redis as a key-value store with TTL-based expiry.
2. Perform CRUD operations in MongoDB; create indexes and analyze query performance using explain().
3. Implement a MapReduce job in MongoDB for aggregating sales data across product categories.

UNIT IV	COLUMN-FAMILY STORES AND GRAPH DATABASES	6+6

Column-Family Stores – What Is a Column-Family Data Store – Features – Consistency – Transactions – Availability – Query Features – Scaling – Suitable Use Cases – Event Logging – Content Management Systems, Blogging Platforms – Counters – Expiring Usage – Graph Databases – What Is a Graph Database – Features – Consistency – Transactions – Availability – Query Features – Scaling – Suitable Use Cases.

List of Exercises:

1. Design and implement a column-family data model using Apache Cassandra for an event logging application.
2. Build a recommendation engine using Neo4j graph queries with Cypher to traverse friend-of-a-friend relationships.

3. Compare query performance between MongoDB (document), Cassandra (column-family), and Neo4j (graph) for the same dataset.

UNIT V	SCHEMA MIGRATIONS, POLYGLOT PERSISTENCE, AND CHOOSING YOUR DATABASE	6+6

Schema Migrations – Schema Changes – Schema Changes in Schemaless Databases – Polyglot Persistence – Polyglot Data Store for an Application – Service Usage over Direct Data Store Usage – Expanding Data Store Usage – Beyond NoSQL – File Systems – Event Sourcing – Memory Image – Choosing Your Database.

List of Exercises:

1. Implement schema migration in MongoDB by evolving a document structure across versions and handling backward compatibility.
2. Mini-Project: Design and develop a polyglot persistence solution for an e-commerce application using MongoDB (product catalog), Redis (session/cart), and Neo4j (recommendations).
3. Case Study: Benchmark and evaluate MongoDB, Cassandra, and Redis for a given use case; present a justified recommendation for database selection.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

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

CO1: Explain the emergence of NoSQL databases, aggregate data models, and the differences between relational and NoSQL approaches. (K2)

CO2: Describe distribution models including sharding and replication, and explain consistency, the CAP theorem, and quorum-based strategies. (K2)

CO3: Apply map-reduce computations and version stamps for data processing and concurrency management in NoSQL databases. (K3)

CO4: Analyze the features, consistency models, and suitable use cases of key-value, document, column-family, and graph databases. (K4)

CO5: Design polyglot persistence architectures and implement schema migration strategies for evolving NoSQL applications. (K5)

CO6: Evaluate and choose appropriate NoSQL database technologies based on application requirements, scalability needs, and performance benchmarks. (K6)

TEXT BOOKS:

1. Sadalage, P. J. and Fowler, M., "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence," Pearson Education (Addison-Wesley), 2013.

2. Bradshaw, S. and Chodorow, K., "MongoDB: The Definitive Guide: Powerful and Scalable Data Storage," 3rd Edition, O'Reilly Media, 2019.

REFERENCES:

1. Meier, A. and Kaufmann, M., "SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management," Springer, 2019.

2. Perkins, L., Redmond, E., and Wilson, J., "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement," 2nd Edition, Pragmatic Bookshelf, 2018.

3. https://www.ibm.com/cloud/learn/nosql-databases

4. https://www.geeksforgeeks.org/introduction-to-nosql/

5. MongoDB Official Documentation: https://www.mongodb.com/docs/

LIST OF EQUIPMENTS:

Computer/Laptop with minimum 4GB RAM, recommended 8GB+ RAM.

Processor: Intel i3/i5/i7 or AMD equivalent.

Storage: At least 50GB free disk space, SSD recommended.

Operating System: Windows, macOS, or Linux (Ubuntu recommended).

NoSQL Database Software: MongoDB, Cassandra, Redis, Neo4j.

NoSQL Management Tools: MongoDB Compass, RedisInsight, Neo4j Browser.

Programming Language Support: Python or Node.js with NoSQL drivers.

Text Editor/IDE: VS Code, PyCharm, or any suitable editor.

24AM904	ADVANCED DATABASES	L	T	P	C
		2	0	2	3

OBJECTIVES: The Course will enable learners to:

- Learn advanced SQL features including complex queries, triggers, views, and SQL programming techniques.
- Understand the concepts of distributed databases, data fragmentation, replication, and distributed query processing.
- Study object and object-relational database concepts and XML data models.
- Learn NoSQL systems and big data technologies including MapReduce and Hadoop.
- Explore advanced database models including active, temporal, spatial, multimedia, and deductive databases.
- Gain practical experience in implementing advanced database features using open-source tools.

UNIT I	ADVANCED SQL AND SQL PROGRAMMING TECHNIQUES	6+6

More SQL: Complex Queries, Triggers, Views, and Schema Modification: More Complex SQL Retrieval Queries - Specifying Constraints as Assertions and Actions as Triggers - Views (Virtual Tables) in SQL - Schema Change Statements in SQL.

Introduction to SQL Programming Techniques: Database Programming: Techniques and Issues - Embedded SQL - Dynamic SQL - Database Programming with Function Calls: SQL/CLI and JDBC.

List of Exercises / Experiments:

1. Write complex SQL queries using nested subqueries, correlated subqueries, EXISTS, GROUP BY, and HAVING on a sample database (e.g., COMPANY or UNIVERSITY schema) using MySQL or PostgreSQL.
2. Create SQL triggers and assertions on the sample database to enforce business rules; test trigger execution on INSERT, UPDATE, and DELETE operations.
3. Write a simple database application using JDBC (Java) or a Python DB-API connector to perform CRUD operations on a relational database.

UNIT II	DISTRIBUTED DATABASES AND CLIENT–SERVER ARCHITECTURES	6+6

Distributed Databases and Client–Server Architectures : Distributed Database Concepts - Types of Distributed Database Systems - Distributed Database Architectures - Data Fragmentation - Overview of Concurrency Control and Recovery in Distributed Databases - Distributed Query Processing - Overview of Three-Tier Client–Server Architecture.

List of Exercises / Experiments:

4. Set up a distributed database environment using MySQL Replication or PostgreSQL logical replication; create horizontally and vertically fragmented tables across two database instances.
5. Write distributed queries that retrieve data from multiple fragments; compare query execution plans of local vs. distributed joins.
6. Simulate a concurrency control scenario: execute concurrent transactions on replicated data and observe conflict resolution behaviour.

UNIT III	OBJECT AND OBJECT-RELATIONAL DATABASES AND XML	6+6

Object and Object-Relational Databases: Overview of Object Database Concepts - Object-Relational Features: SQL Structured Types, Typed Tables, and Object-Relational Features in SQL - The ODMG Object Model and the Object Definition Language - Object Query Language.

XML: Extensible Markup Language: Structured, Semistructured, and Unstructured Data - XML Hierarchical Data Model - XML Query Languages - Extracting XML Documents from Relational Databases.

List of Exercises / Experiments:

7. Create user-defined types and typed tables in PostgreSQL using SQL object-relational

features (CREATE TYPE, inheritance); write OQL-style queries.

8. Design an XML schema (XSD) for a sample domain (e.g., Library or Student Records); validate XML documents against the schema.
9. Write XPath and XQuery expressions to retrieve data from XML documents; use SQL/XML functions to extract XML from a relational database.

UNIT IV	NOSQL SYSTEMS AND BIG DATA TECHNOLOGIES	6+6

Introduction to NOSQL Systems: Introduction to NOSQL Systems - Document-Based NOSQL Systems and MongoDB - NOSQL Key-Value Stores - Column-Based or Wide Column NOSQL Systems - NOSQL Graph Databases and Neo4j.

Big Data Technologies Based on MapReduce and Hadoop: Big Data Technologies - The MapReduce Programming Model - The Hadoop Framework - Higher-Level Interfaces for Hadoop.

List of Exercises / Experiments:

10. Install MongoDB (open source); create collections, insert documents with nested structures, and write queries using find(), aggregate pipeline, and indexing.
11. Model a social network or recommendation system using Neo4j (open source graph database); write Cypher queries for traversal and pattern matching.
12. Write a simple MapReduce program (word count or log analysis) using Hadoop Streaming or Apache Spark (open source); run it on a local cluster or single-node setup.

UNIT V	ENHANCED DATA MODELS, INFORMATION RETRIEVAL AND WEB SEARCH	6+6

Enhanced Data Models: Introduction to Active, Temporal, Spatial, Multimedia, and Deductive Databases: Active Database Concepts and Triggers - Temporal Database Concepts - Spatial Database Concepts - Multimedia Database Concepts - Deductive Database Concepts.

Introduction to Information Retrieval and Web Search: Information Retrieval (IR) Concepts - Retrieval Models - Search Evaluation - Web Search and Ranking.

List of Exercises / Experiments:

13. Use PostGIS (open-source spatial extension for PostgreSQL) to create spatial tables, insert geographic data (points, polygons), and write spatial queries (ST_Contains, ST_Distance, ST_Within).
14. Implement a mini information retrieval system: build an inverted index from a text corpus using Python (open source), compute TF–IDF scores, and rank documents for a given query.
15. Mini-project / Case study: Design and implement an integrated database application that combines at least two advanced database technologies (e.g., a location-aware recommendation system using PostgreSQL + PostGIS + MongoDB, or a document QA system using full-text search + Neo4j knowledge graph). Include a written report comparing the relational and NoSQL approaches used.

TOTAL: 30 + 30 = 60 PERIODS

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

CO1: Write complex SQL queries using nested subqueries, triggers, views, and SQL programming techniques. *[K3 – Apply]*

CO2: Explain the architecture of distributed databases and the techniques for data fragmentation, replication, and distributed query processing. *[K2 – Understand]*

CO3: Apply object-relational database features and XML query languages to model and query semi-structured data. *[K3 – Apply]*

CO4: Analyze the trade-offs between relational, NoSQL, and big data systems using the CAP theorem and choose appropriate storage models. *[K4 – Analyze]*

CO5: Design and implement database solutions using advanced data models such as spatial, temporal, and graph databases. *[K6 – Create]*

CO6: Evaluate information retrieval techniques and build search systems using inverted indexes, TF–IDF, and ranking algorithms. *[K5 – Evaluate]*

TEXT BOOKS:

1. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson, 2016. ISBN: 978-0-13-397077-7.

REFERENCES:

1. Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems", 3rd Edition, McGraw-Hill, 2003. ISBN: 978-0-07-246563-1.
2. S. Ceri and G. Pelagatti, "Distributed Databases: Principles and Systems", McGraw-Hill, 1984. ISBN: 978-0-07-010829-1.
3. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, "Database System Concepts", 7th Edition, McGraw-Hill, 2020. ISBN: 978-0-07-802215-9.
4. Pramod J. Sadalage and Martin Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Addison-Wesley, 2013. ISBN: 978-0-321-82662-6.
5. MongoDB Documentation. Freely available at https://www.mongodb.com/docs/
6. Neo4j Documentation. Freely available at https://neo4j.com/docs/
7. Apache Hadoop Documentation. Freely available at https://hadoop.apache.org/docs/

24AM905	IMAGE AND VIDEO ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the basics of image processing techniques for computer vision and video analysis.
- To illustrate the techniques used for image pre-processing.
- To discuss the various image Segmentation techniques.
- To understand the various Object recognition mechanisms.
- To elaborate on the motion analysis techniques for video analytics.

UNIT I	INTRODUCTION	9

Computer Vision – Image representation and image analysis tasks - Image representations – digitization – properties – color images – Data structures for Image Analysis - Levels of image data representation - Traditional and Hierarchical image data structures.

UNIT II	IMAGE PRE-PROCESSING	9

Pixel brightness transformations - Geometric transformations - Local pre-processing - Image smoothing - Edge detectors - Zero-crossings of the second derivative - Scale in image processing - Canny edge detection - Parametric edge models - Edges in multi-spectral images - Local pre-processing in the frequency domain - Line detection by local pre-processing operators - Detection of corners (interest points) - Detection of maximally stable extremal regions - Image restoration.

UNIT III	SEGMENTATION	9

Thresholding - Edge-based segmentation - Region-based segmentation – Matching - Evaluation issues in segmentation - Mean shift segmentation - Active contour models.

UNIT IV	OBJECT RECOGNITION	9

Knowledge representation - Statistical pattern recognition - Neural nets - Syntactic pattern recognition - Recognition as graph matching - Optimization techniques in recognition - Fuzzy systems - Boosting in pattern recognition - Random forests - Image understanding control strategies.

UNIT V	MOTION ANALYSIS	9

Differential motion analysis methods - Optical flow - Analysis based on correspondence of interest points - Detection of specific motion patterns - Video tracking - Motion models to aid tracking.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics of image processing techniques for computer vision and video analysis.
CO2: Illustrate the techniques used for image pre-processing.
CO3: Analyze the various image Segmentation techniques.
CO4: Understand the various Object recognition mechanisms.
CO5: Elaborate on the motion analysis techniques for video analytics.
CO6: Apply image processing techniques in real-world applications.

TEXT BOOKS:

1. Milan Sonka, Vaclav Hlavac, Roger Boyle, “Image Processing, Analysis, and Machine Vision”, 4nd edition, Thomson Learning, 2013.

REFERENCES:

1. Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer Verlag LondonLimited, 2011.
2. Caifeng Shan, Fatih Porikli, Tao Xiang, Shaogang Gong, “Video Analytics for BusinessIntelligence”, Springer, 2012.
3. D. A. Forsyth, J. Ponce, “Computer Vision: A Modern Approach”, Pearson Education, 2003.
4. E. R. Davies, “Computer & Machine Vision”, Fourth Edition, Academic Press,2012.

24AM906	STREAM PROCESSING AND ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To outline the framework for real time stream processing.
- To learn various algorithms for data streaming.
- To identify frequent item sets by mining from data streams.
- To introduce approaches to evaluate stream learning algorithms.
- To use tools for distributed data flow management.
- To design solutions to stream processing problems

UNIT I	INTRODUCTION TO DATA STREAMS	9

Data Stream Models – Bounds of Random variables – Poisson Process – Maintaining Simple Statistics from Data Streams – Sliding Window and computing statistics over sliding windows – Data Synopsis – Sampling – Histograms – Wavelets – DFT - Change Detection: Tracking Drifting Concepts - Monitoring the Learning Process.

UNIT II	STREAMING ALGORITHMS	9

Clustering Examples: Basic Concepts - Partitioning Clustering – Hierarchical Clustering - Micro Clustering – Grid Clustering - Clustering Variables - The Very Fast Decision Tree Algorithm (VFDT) - The Base Algorithm, Analysis of the VFDT Algorithm, Extensions to the Basic Algorithm: Processing Continuous Attributes, Functional Tree Leaves, Concept Drift.

UNIT III	FREQUENT PATTERN MINING	9

Introduction – Heavy Hitters - Mining Frequent Itemsets from Data Streams - Landmark Windows - Mining Recent Frequent Itemsets - Frequent Itemsets at Multiple Time Granularities - Sequence Pattern Mining - Reservoir Sampling for Sequential Pattern Mining over data stream.

UNIT IV	EVALUATING STREAMING ALGORITHMS	9

Learning from Data Streams - Evaluation Issues - Design of Evaluation Experiments - Evaluation Metrics - Comparative Assessment - Evaluation Methodology in Non-Stationary Environments.

UNIT V	DATA FLOW MANAGEMENT	9

Distributed Data Flows – Apache Kafka – Apace Flume - Processing Streaming Data – Storing Streaming Data – Delivering Streaming Metrics.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: To outline the framework for real time stream processing.

CO2: To learn various algorithms for data streaming.

CO3: To identify frequent item sets by mining from data streams.

CO4: To introduce approaches to evaluate stream learning algorithms.

CO5: To use tools for distributed data flow management.

CO6: To design solutions to stream processing problems

TEXT BOOKS:

1. Joao Gama, "Knowledge Discovery from Data Streams", CRC Press, 2010.
2. Byron Ellis, Real-Time Analytics: Techniques to Analyze and Visualize Streaming Data, First Edition, WILEY Big Data Series, 2014

REFERENCES:

1. Andrew Psaltis, Streaming Data: Paul Lewis, First Edition, Manning Publication, 2017.
2. Bugra Gedik, Deepak S. Turaga, Henrique C. M. Andrade, Fundamentals of Stream Processing: Application Design, Systems, and Analytics, Cambridge University Press, 2014.

3. Charu C. Aggarwal, “Data Streams: Models and Algorithms”, Kluwer Academic Publishers, 2007.
4. David Luckham, “The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems”, Addison Wesley, 2002.
LIST OF EQUIPMENTS:
Apache Flink, Python API via PyFlink, Apache Kafka, SQLite, TimescaleDB , Grafana

24AM907	SOCIAL NETWORK ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To outline the components of the social network.
- To elaborate the modeling and visualization of the social network.
- To classify descriptive and inferential methods.
- To discuss about the evolution of the social network.
- To illustrate the applications in real time systems.

UNIT I	INTRODUCTION	9

Basics of Social Network Analysis: Introduction- The Social network and Representation -Types of Networks-Network parts and Level of Analysis-Networks as Social Structure and Institution-Theoretical Assumptions-Causality in Social Network Studies- A Brief History of Social Network Analysis-Mathematical Foundations: Graphs-Paths and components-Adjacency matrices-Ways and modes-Matrix products-Sources of network data-Types of nodes and types of ties- Data Collection: Network questions-Question formats-Interviewee burden-Data collection and reliability-Archival data collection-Data from electronic sources.

UNIT II	MODELING AND VISUALIZATION	9

Data Management: Data import-Cleaning network data- Data transformation-Normalization-Cognitive social structure data-Matching attributes and networks-Converting attributes to matrices-Data export,- Multivariate Techniques Used in Network Analysis: Multidimensional scaling-Correspondence analysis-Hierarchical clustering,- Visualization: Layout-Embedding node attributes-Node filtering-Ego networks-Embedding tie characteristics-Visualizing network change-Exporting visualizations-Closing comments.

UNIT III	DESCRIPTIVE AND INFERENTIAL METHODS	9

Descriptive Methods in Social Network Analysis: Graph and Matrix-Social Network Representation – Density – Centrality, Centralization and Prestige- Cliques – Multidimensional Scaling(MDS) and Dendogram – Structural Equivalence -Two mode Networks and Bipartite Matrix-Inferential Methods in Social Network Analysis: Permutation and QAP (Quadratic Assignment Procedure) Correlation-P* or Exponential Random Graph Model(ERGM).

UNIT IV	EVOLUTION	9

Evolution in Social Networks – Framework - Tracing Smoothly Evolving Communities - Models and Algorithms for Social Influence Analysis - Influence Related Statistics - Social Similarity and Influence - Influence Maximization in Viral Marketing - Algorithms and Systems for Expert Location in Social Networks - Expert Location without Graph Constraints - with Score Propagation – Expert Team Formation - Link Prediction in Social Networks - Feature based Link Prediction – Bayesian Probabilistic Models - Probabilistic Relational Models.

UNIT V	APPLICATIONS	9

A Learning Based Approach for Real Time Emotion Classification of Tweets, A New Linguistic Approach to Assess the Opinion of Users in Social Network Environments, Explaining Scientific and Technical Emergence Forecasting, Social Network Analysis for Biometric Template Protection

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Outline the internal components and terminology of the social network.
CO2: Illustrate the fundamental exploratory multivariate techniques and visualizing network data.
CO3: Discuss most common descriptive and inferential statistical tools available.
CO4: Discuss about the evolution of the social network.
CO5: Illustrate the real time applications of social network analysis.
CO6: Apply the methods in Social Network Analysis to solve real world problems.

TEXT BOOKS:

1. Song Yang , Franziska B. Keller, “Social Network Analysis Methods and Examples”, SAGE Publications Inc. 2017.
2. Stephen P Borgatti, Martin G. Everett, Jeffrey C. Johnson, “Analyzing Social Networks”,

Second Edition, 2017.

REFERENCES:

1. Charu C. Aggarwal, “Social Network Data Analytics”, Springer; 2014 .
2. Przemyslaw Kazienko, Nitesh Chawla, “Applications of Social Media and Social Network Analysis”, Springer,2015.
3. Ajith Abraham, Aboul Ella Hassanien, Vaclav Snasel, “Computational Social Network Analysis: Trends, Tools and Research Advances”, Springer, 2012.
4. Borko Furht, “Handbook of Social Network Technologies and Applications”, Springer, 1st edition, 2011.
5. Guandong Xu , Yanchun Zhang and Lin Li, “Web Mining and Social Networking – Techniques and applications”, Springer, 1st edition, 2012.

24AM908	COGNITIVE SCIENCE AND ANALYTICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand cognitive computing.
- To know about design principles and NLP for Cognitive systems.
- To distinguish between Big Data and Cognitive computing.
- To discuss implications of cognitive computing in business.
- To develop applications of cognitive computing.

UNIT I	FOUNDATIONS OF COGNITIVE SCIENCE	9

Foundation of Cognitive Computing: cognitive computing as a new generation- the uses of cognitive systems- system cognitive- gaining insights from data- Artificial Intelligence as the foundation of cognitive computing- understanding cognition.

UNIT II	DESIGN PRINCIPLES FOR COGNITIVE SYSTEMS AND NLP IN COGNITIVE SYSTEMS	9

Components of a cognitive system- building the corpus- bringing data into cognitive system- machine learning- hypotheses generation and scoring- presentation and visualization services.
Natural Language Processing in support of a Cognitive System: Role of NLP in a cognitive system- semantic web- Applying Natural language technologies to Business problems.

UNIT III	BIG DATA Vs COGNITIVE COMPUTING	9

Relationship between Big Data and Cognitive Computing: Dealing with human-generated data- defining big data- architectural foundation- analytical data warehouses- Hadoop- data in motion and streaming data- integration of big data with traditional data.

UNIT IV	THE BUSINESS IMPLICATIONS OF COGNITIVE COMPUTING	9

Preparing for change- advantages of new disruptive models- knowledge meaning to business- difference with a cognitive systems approach- meshing data together differently- using business knowledge to plan for the future- answering business questions in new ways- building business specific solutions- making cognitive computing a reality- cognitive application changing the market- IBM Watson as a cognitive system.

UNIT V	APPLICATIONS OF COGNITIVE COMPUTING	9

Build a cognitive health care application - Build a cognitive application on Smarter cities - Applicate Cognitive Computing principle in building a Government related application.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the concepts of cognitive science and computing.
CO2: Design and Implementation of Cognitive Systems.
CO3: Apply NLP in cognitive systems.
CO4: Integrate Big Data and Cognitive computing.
CO5: Discuss implications of cognitive computing in business.
CO6: Develop various applications of cognitive computing.

TEXT BOOKS:

1. Judith H Hurwitz, Marcia Kaufman, Adrian Bowles, “Cognitive computing and Big Data Analytics” , Wiley, 2015.

REFERENCES:

1. Vijay Raghvan, Venu Govindaraju, C.R. Rao, “Cognitive Computing: Theory and Applications“, Elsevier publications, North Holland Publication, 1st Edition, 2016.
2. Mallick, Pradeep Kumar, Borah, Samarjeet, "Emerging Trends and Applications in Cognitive Computing”, IGI Global Publishers, 2019.

24AM909	Web Analytics	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the concepts and evolution of Web Analytics.
- To analyze website data using metrics and key performance indicators (KPIs).
- To explore qualitative and quantitative analysis techniques.
- To apply testing, experimentation, and competitive intelligence methods.
- To study emerging trends and develop data-driven decision-making skills.

UNIT I	INTRODUCTION TO WEB ANALYTICS 2.0	9

Introduction – Evolution of Web Analytics – State of Analytics Industry – Web Analytics 2.0 concept – Clickstream data – Multiple outcomes analysis – Experimentation and testing – Voice of customer – Competitive intelligence – Strategic importance of analytics – Tactical implementation – Selection of web analytics tools – Vendor evaluation and tool selection strategies.

UNIT II	CLICKSTREAM ANALYSIS AND METRICS	9

Clickstream analysis – Key web metrics: Visits, Visitors, Time on page, Bounce rate, Exit rate, Conversion rate, Engagement – Characteristics of good metrics – Diagnosing metric performance – Traffic source analysis – Customer behavior analysis – Segmentation techniques – SEO, PPC, Email and Direct traffic analysis – Custom reporting and insights.

UNIT III	MEASURING SUCCESS AND KPI ANALYSIS	9

Key Performance Indicators (KPIs) – Macro and Micro conversions – Task completion rate – Visitor loyalty and recency – Conversion funnel analysis – Cart abandonment – Purchase behavior – Average order value – Measuring success for e-commerce, non e-commerce and B2B websites – Economic value of analytics.

UNIT IV	ADVANCED ANALYTICS AND OPTIMIZATION TECHNIQUES	9

Qualitative analytics – Usability testing – Surveys and customer feedback – A/B testing and multivariate testing – Experimentation techniques – Competitive intelligence analysis – Traffic and keyword analysis – Audience segmentation – Benchmarking – Market analysis strategies.

UNIT V	EMERGING TRENDS AND ANALYTICS STRATEGY	9

Social media analytics – Mobile analytics – Video analytics – Data quality and dashboard design – Multichannel analytics – Attribution models – Predictive analytics basics – Web analytics career – Building data-driven culture – Challenges and future trends in analytics.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the concepts and evolution of Web Analytics and Web Analytics 2.0.
CO2: Analyze website performance using key metrics and clickstream data.
CO3: Apply KPIs and conversion analysis to measure business success.
CO4: Evaluate qualitative and quantitative techniques including testing and experimentation..
CO5: Assess competitive intelligence and advanced analytics strategies.
CO6: Examine emerging trends and develop data-driven decision-making skills.

TEXT BOOK:

1. Web Analytics 2.0, "Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity", Wiley Publishing.

REFERENCES:

1. Web Analytics: An Hour a Day, “Web Analytics: An Hour a Day”, Wiley Publishing.
2. Understanding User-Web Interactions via Web Analytics, “Understanding User-Web Interactions via Web Analytics”, Springer.
3. Social Media and Web Analytics, “Social Media and Web Analytics”, McGraw Hill (Latest Edition).

APPLIED AI

24AM910	AI IN AGRICULTURE AND PRECISION FARMING	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

1. Understand the fundamentals of precision agriculture and the role of AI in modern farming.
2. Learn machine learning and deep learning techniques applied to crop, soil, and water management.
3. Explore sensor technologies, IoT, remote sensing, and drone-based systems for agricultural data acquisition.
4. Study AI-driven decision support systems for farm management and resource optimization.
5. Analyze real-world case studies and emerging trends of AI adoption in agriculture.

UNIT I	INTRODUCTION TO AI IN AGRICULTURE AND PRECISION FARMING	9

Evolution of Digital Agriculture – Traditional Farming vs. Smart Farming – Role of AI and Big Data in Smart Farming – Overview of Precision Agriculture – A History of Precision Agriculture – Key Components of Precision Agriculture – Sensing Technology for Precision Crop Farming – GPS, GIS, and Remote Sensing in Agriculture – Challenges and Opportunities of AI Adoption in Agriculture.

UNIT II	MACHINE LEARNING FOR CROP AND SOIL MANAGEMENT	9

Machine Learning Techniques for Agriculture – Supervised and Unsupervised Learning for Crop Management – Crop Yield Prediction using Machine Learning – Crop Disease Detection and Classification – Weed Detection and Crop Quality Assessment – Soil Health Monitoring and Soil Property Prediction – Data Processing and Utilization in Precision Agriculture – Decision Support Systems for Precision Nutrient Management.

UNIT III	DEEP LEARNING AND COMPUTER VISION IN AGRICULTURE	9

Convolutional Neural Networks for Plant Disease Detection – Image Classification and Object Detection in Agricultural Imaging – Deep Learning in Agricultural Imaging for Crop Health Monitoring – Transfer Learning for Crop and Weed Identification – Satellite Imagery and Geospatial Data Analytics for Agriculture – Drone-based Crop Health Monitoring – Spectral Imaging and NDVI Analysis – Deep Learning for Yield Forecasting and Estimation.

UNIT IV	IOT, SENSORS, AND SMART FARMING SYSTEMS	9

IoT-enabled Smart Farming Systems – Wireless Sensor Networks for Agricultural Monitoring – Soil Moisture Sensors and Weather Stations – Smart Irrigation and Water Management using AI – Variable Rate Technology for Precision Input Application – Intelligent Agricultural Machinery and Field Robots – Autonomous Tractors and Robotic Harvesters – Control of Precision Agriculture Production – Integration of IoT with Big Data Platforms for Farm Analytics.

UNIT V	AI FOR SUSTAINABLE AGRICULTURE AND EMERGING TRENDS	9

Predictive Analytics for Pest and Disease Management – AI in Climate-Smart Agriculture – Weather and Climate Risk Analytics for Early Warning Systems – AI for Supply Chain Optimization and Traceability – Livestock Management using AI and Sensor Data – Ethical Considerations and Data Privacy in Digital Agriculture – AI-Driven Farmer Advisory and Extension Services – Emerging Trends: Generative AI, LLMs, and Foundation Models in Agriculture.

TOTAL: 45 PERIODS

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

CO1: Describe the fundamentals of precision agriculture and the significance of AI technologies in modern farming systems. (K2)

CO2: Apply machine learning algorithms for crop yield prediction, disease detection, and soil health assessment. (K3)

CO3: Analyze agricultural image data using deep learning and computer vision techniques for crop health

monitoring and weed identification. (K4)

CO4: Design IoT-based smart farming solutions integrating sensors, drones, and intelligent machinery for precision resource management. (K5)

CO5: Evaluate AI-driven decision support systems for sustainable agriculture, climate adaptation, and supply chain optimization. (K5)

CO6: Develop an AI-based agricultural application by integrating machine learning, sensor data, and domain knowledge for a real-world farming problem. (K6)

TEXT BOOKS:

1. Rajesh Singh, Anita Gehlot, Mahesh Kumar Prajapat, Bhupendra Singh, "Artificial Intelligence in Agriculture", CRC Press (Taylor & Francis), 1st Edition, 2021. (ISBN: 9781032158105)

2. Qin Zhang (Ed.), "Precision Agriculture Technology for Crop Farming", CRC Press (Taylor & Francis), 1st Edition, 2015. (ISBN: 9781032098272)

REFERENCES:

1. Mohd. Anul Haq, Ilyas Khan (Eds.), "Application of Machine Learning in Agriculture", Elsevier, 1st Edition, 2022. (ISBN: 9780323905503)

2. Swain, K.C., "A Textbook on Precision Agriculture", New Delhi Publishers, 2020.

3. Konstantinos G. Liakos et al., "Machine Learning in Agriculture: A Review", Sensors, Vol. 18, No. 8, 2674, MDPI, 2018. (Open Access)

4. NPTEL Course: "Precision Agriculture" by Dr. Kishore Chandra Swain, IIT Kharagpur. (Available at: https://onlinecourses.nptel.ac.in)

5. Sara Oleiro Araujo et al., "Machine Learning Applications in Agriculture: Current Trends, Challenges, and Future Perspectives", Agronomy, Vol. 13, No. 12, 2976, MDPI, 2023. (Open Access)

24AM911	AI in BLOCK CHAIN	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To acquire knowledge in Blockchain Technologies.
- To understand how block chain and AI can be used to innovate.
- To elaborate Cryptocurrencies and AI.
- To develop applications using blockchain.
- To understand the limitations and future scope of AI in Blockchain.

UNIT I	INTRODUCTION TO BLOCKCHAIN	9

Overview – Blockchain vs Distributed Ledger Technology vs Distributed Databases – Public vs private vs permissioned blockchains – Privacy in blockchains – Blockchain platforms - Hyperledger – Hashgraph, Corda – IOTA - Consensus Algorithms – Building DApps with blockchain tools.

UNIT II	BLOCKCHAIN AND ARTIFICIAL INTELLIGENCE	9

Introduction to the AI landscape - AI and Blockchain driven Databases – Centralized vs Distributed data – Blockchain data – Big data for AI analysis – Global databases – Data Management in a DAO - Benefits of combining blockchain and AI – Aicumen Technologies -Combining blockchain and AI to humanize digital interactions.

UNIT III	CRYPTOCURRENCY AND AI	9

Bitcoins – Ethereum - Role of AI in cryptocurrency – cryptocurrency trading – Making price predictions with AI – Market making – future of cryptocurrencies.

UNIT IV	DEVELOPING BLOCKCHAIN PRODUCTS	9

Development Life Cycle of a DIApp – Designing a DIApp – Developing a DIApp – Testing – Deploying – Monitoring – Implementing DIApps.

UNIT V	LIMITATIONS AND FUTURE OF AI WITH BLOCKCHAIN	9

Technical Challenges – Business Model Challenges – Scandals and Public perception – Government Regulation – Privacy Challenges for Personal Records – Convergence of AI with Blockchain – Future – Enterprise.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Acquire knowledge in Blockchain Technologies.
CO2: Understand how block chain and AI can be used to innovate.
CO3: Elaborate Cryptocurrencies and AI.
CO4: Develop applications using blockchain.
CO5: Understand the limitations and future scope of AI in Blockchain.
CO6: Elaborate the various applications of AI in Blockchain.

TEXT BOOKS:

1. Ganesh Prasad Kumble, Anantha Krishnan, "Practical Artificial Intelligence and Blockchain: A guide to converging blockchain and AI to build smart applications for new economies", Packt Publications, 2020.
2. Melanie Swan, "Block Chain: Blueprint for a New Economy", O'Reilly, 2015.

REFERENCES:

1. Daniel Drescher, "Block Chain Basics", Apress; 1st edition, 2017.
2. Salman Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O'Dowd, Venkatraman Ramakrishna, "Hands-On Block Chain with Hyperledger: Building Decentralized Applications with Hyperledger Fabric and Composer", Import, 2018.

24AM912	AI IN CYBER SECURITY	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Learn the Concepts of AI in Cybersecurity.
- Understand the methods of Cybersecurity Arsenal
- Learn the techniques for security threats with AI
- Protect the sensitive information and assets.
- Explore the Tools and Applications to learn about the AI in Cybersecurity

UNIT I	INTRODUCTION TO AI IN CYBERSECURITY	9

Introduction – Applying AI in cyber security - Evolution from expert systems to data mining and AI- The different forms of automated learning - Characteristics of algorithm training and optimization - Beginning with AI via Jupyter Notebooks - Introducing AI in the context of cyber security.

UNIT II	AI FOR CYBER SECURITY ARSENAL	9

Classification – Regression - Dimensionality reduction – Clustering - Speech recognition - Video anomaly detection - Natural language processing (NLP) - Large- scale image processing - Social media analysis.

UNIT III	CYBER SECURITY THREATS WITH AI	9

Detection of Spam with Perceptron - Image spam detection with support vector machines - Phishing detection with logistic regression and decision trees- Spam detection with Naive Bayes - Spam detection adopting NLP.

UNIT IV	PROTECTING SENSITIVE INFORMATION AND ASSETS	9

Authentication abuse prevention - Account reputation scoring - User authentication with keystroke recognition -Biometric authentication with facial recognition.

UNIT V	APPLICATIONS	9

Machine learning (ML) algorithms for fraud detection - bagging and boosting techniques - analyze data with IBM Watson and Jupyter Notebook - resort to statistical metrics for results evaluation.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop security applications of social networks.
CO2: Implement AI for cyber security arsenal.
CO3: Analyze and handle security threats in AI
CO4: To Protect and handle sensitive information and assets.
CO5: Develop applications for AI for fraud detection.
CO6: Create newer AI application in support with cybersecurity.

TEXT BOOK:

2. Alessandro Parisi .(2019). Hands-On Artificial Intelligence for Cybersecurity: Implement smart AI systems for preventing cyber attacks and detecting threats and network anomalies, Packt Publication1

REFERENCES:

1. Mark Stamp Corrado Aaron Visaggio Francesco Mercaldo Fabio Di Troia Editors, " Artificial Intelligence for Cybersecurity" , Springer7
2. Leslie F. Sikos Editor "AI in Cybersecurity", Springer

24AM913	ETHICAL AND RESPONSIBLE AI	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- Understand the societal impact of artificial intelligence and the need to discuss ethical and societal problems raised by AI.
- Learn the philosophical foundations concerning the differences between humans and machines and the moral status of AI.
- Explore the key ethical issues in AI including privacy, responsibility, transparency, explainability, and bias.
- Study the role of data science and machine learning in creating ethical challenges across AI applications.
- Examine policy proposals, governance frameworks, and responsible innovation practices for ethical AI deployment.
- Evaluate the broader societal challenges of AI including the future of work, sustainability, and climate implications.

UNIT I	AI NARRATIVES AND PHILOSOPHICAL FOUNDATIONS (Ch 1–4)	9

Mirror, Mirror, on the Wall: The AI Hype and Fears – The Real and Pervasive Impact of AI – The Need to Discuss Ethical and Societal Problems.
Superintelligence, Monsters, and the AI Apocalypse: Superintelligence and Transhumanism – Frankenstein's New Monster – Transcendence and the AI Apocalypse – How to Go beyond Competition Narratives and beyond the Hype – Is General AI Possible?
All about the Human: What Makes Humans Different from Machines? – Consciousness, Emotion, and Embodiment – The Turing Test and Its Limitations – Moral Status and Moral Agency.
Just Machines? Can Machines Be Moral? – Machine Ethics and Moral Machines – Moral Status of Robots and AI – Social Robots and Human–Robot Relations.

UNIT II	AI TECHNOLOGY AND DATA SCIENCE (Ch 5–6)	9

The Technology: A Short History of AI – What Is AI? Definitions and Distinctions – Symbolic AI and Connectionism – Different Approaches and Subfields – Deep Learning and Neural Networks – Natural Language Processing – Computer Vision – Robotics – Applications and Impact.
Don't Forget the Data (Science): Machine Learning – Supervised, Unsupervised, and Reinforcement Learning – Data Science – The Role of Data in AI – Data Collection, Data Quality, and Data Curation – Bias in Data and Algorithms – Applications of Data Science – Ethical Challenges in Data Science.

UNIT III	PRIVACY, RESPONSIBILITY, AND TRANSPARENCY (Ch 7–8)	9

Privacy and the Other Usual Suspects: Privacy and Data Protection – Surveillance and Power – Manipulation, Exploitation, and Vulnerable Users – Fake News, the Danger of Totalitarianism, and the Impact on Personal Relations – Safety and Security Risks – Autonomous Vehicles and Lethal Autonomous Weapons.
A-responsible Machines and Unexplainable Decisions: Responsibility and Accountability – The Problem of Many Hands – Delegation of Decision Making – The Responsibility Gap – Transparency and Explainability – Black Box Problem – Explainable AI (XAI) – Trust and AI Systems.

UNIT IV	BIAS, FAIRNESS, AND THE FUTURE OF WORK (Ch 9)	9

Bias and the Meaning of Life: Bias – Sources of Bias in AI – Bias at All Stages of the Data Science Process – Algorithmic Bias and Discrimination – Fairness and Justice – Bias in Predictive Policing, Criminal Justice, and Hiring – Gender Bias and Racial Bias in AI Systems.

The Future of Work and the Meaning of Life: Automation and Job Displacement – AI and the Labor Market – Inequality and the Digital Divide – The Gig Economy and Platform Work – Universal Basic Income – Human Dignity and the Meaning of Work – Education and Reskilling for an AI Economy.

UNIT V	POLICY, GOVERNANCE, AND SUSTAINABILITY (Ch 10–12)	9

Policy Proposals: What Needs to Be Done and Other Questions Policymakers Have to Answer – Ethical Principles and Justifications – Technological Solutions and the Question of Methods and Operationalization – Regulation and Governance Frameworks – EU AI Act – GDPR and Data Protection – Global AI Ethics Guidelines.

Challenges for Policymakers: Proactive Ethics: Responsible Innovation and Embedding Values in Design – Practice Oriented and Bottom Up: How Can We Translate These to Practice? – Toward a Positive Ethics – Interdisciplinarity and Transdisciplinarity – The Risk of an AI Winter and the Danger of the Mindless Use of AI.

It's the Climate, Stupid! Should AI Ethics Be Human-Centric? – Getting Our Priorities Right – AI, Climate Change, and the Anthropocene – Toward Sustainable AI – Wanted: Intelligence and Wisdom.

TOTAL: 45 PERIODS

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

CO1: Explain the key narratives, philosophical foundations, and societal impact of artificial intelligence. *[K2 – Understand]*

CO2: Describe the technology of AI, machine learning approaches, and the role of data science in creating ethical challenges. *[K2 – Understand]*

CO3: Apply ethical principles of privacy, transparency, and explainability to assess real-world AI applications. *[K3 – Apply]*

CO4: Analyze sources of bias and fairness issues in AI systems across the data science pipeline and evaluate their societal consequences. *[K4 – Analyze]*

CO5: Evaluate policy proposals, governance frameworks, and regulatory approaches for responsible AI deployment. *[K5 – Evaluate]*

CO6: Design responsible AI solutions by embedding ethical values, sustainability considerations, and accountability mechanisms into AI systems. *[K6 – Create]*

TEXT BOOKS:

1. Mark Coeckelbergh, "AI Ethics", MIT Press (Essential Knowledge Series), 2020. ISBN: 9780262538190.

REFERENCES:

1. Luciano Floridi, "The Ethics of Artificial Intelligence: Principles, Challenges, and Opportunities", Oxford University Press, 2023. ISBN: 9780198883098.
2. Virginia Dignum, "Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way", Springer, 2019. ISBN: 9783030303709.
3. Solon Barocas, Moritz Hardt, and Arvind Narayanan, "Fairness and Machine Learning: Limitations and Opportunities", MIT Press, 2023. ISBN: 9780262048613. Freely available at https://fairmlbook.org
4. Paula Boddington, "AI Ethics: A Textbook", Springer, 2023. ISBN: 9789811989742.
5. NPTEL Course: "Responsible & Safe AI Systems" by Prof. Ponnurangam Kumaraguru, Prof. Balaraman Ravindran, and Prof. Arun Rajkumar, IIIT Hyderabad & IIT Madras. Available at https://onlinecourses.nptel.ac.in/noc25_cs118

22EC012	INDUSTRIAL IoT APPLICATIONS	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- To introduce how IoT has become a game changer in the new economy where the customers are looking for integrated value.
- To get insights over the architecture and protocols of IIoT
- To know the various sensors and interfacing used in IIoT.
- To bring the IoT perspective in thinking and building solutions.
- To understand the different IoT platforms and cloud services

UNIT I	INTRODUCTION	9

Introduction to IOT, what is IIOT? IOT Vs. IIOT, History of IIOT, Components of IIOT - Sensors, Interface, Networks, People Process, Hype cycle, IOT Market, Trends, future Real-life examples, Key terms – IOT Platform, Interfaces, API, clouds, Data Management Analytics, Mining Manipulation, Thinking about Prototyping – Costs versus ease of prototyping, prototyping and Production, open source versus Closed Source, Role of IIOT in Manufacturing Processes, Use of IIOT in plant maintenance practices, Sustainability through Business excellence tools Challenges, Benefits in implementing IIOT

UNIT II	ARCHITECTURE AND PROTOCOLS	9

Overview of IOT components; Various Architectures of IOT and IIOT, Advantages & disadvantages, Industrial Internet - Reference Architecture; IIOT System components: Sensors, Gateways, Routers, Modem, Cloud brokers, servers and its integration, WSN, WSN network design for IOT; Need for protocols, Wi-Fi, Zigbee, Bacnet, IIOT protocols –COAP, MQTT,6LoWPAN, LWM2M, AMPQ

UNIT III	SENSORS AND INTERFACING	9

Introduction to sensors, Transducers, Classification, Roles of sensors in IIOT, Various types of sensors, Design of sensors, sensor architecture, special requirements for IIOT sensors, Role of actuators, types of actuators. Hardwire the sensors with different protocols such as HART, MODBUS-Serial, Parallel, Ethernet, BACNet , Current, M2M, prototyping online Components– Getting Started with an API, Writing a New API, Real Time Reactions.

UNIT IV	CLOUD, SECURITY AND GOVERNANCE	9

IIOT cloud platforms: Overview of cots cloud platforms, predix, thingworks, azure, Data analytics, cloud services, Business models: Saas, Paas, Iaas; Introduction to web security,Conventional web technology and relationship with IIOT, Vulnerabilities of IoT, IoT security tomography and layered attacker model, Identity establishment, Access control, Message integrity; Management aspects of cyber security.

UNIT V	IOT ANALYTICS AND APPLICATIONS	9

IOT Analytics: Role of Analytics in IOT, Data visualization Techniques, Statistical Methods;IOT Applications: Smart Metering, e-Health Body Area Networks, City Automation,Automotive Applications, Plant Automation, Real life examples of IIOT in Manufacturing Sector.

TOTAL: 45 PERIODS

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

CO1: Describe IOT, IIOT
CO2: Understand various IoT Layers and their relative importance
CO3: Interpret the requirements of IIOT sensors and understand the role of actuators. CO4: Study various IoT platforms and Security
CO5: Realize the importance of Data Analytics in IoT
CO6: Design various applications using IIoT in manufacturing sector.

TEXT BOOKS:

- Daniel Minoli, Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications,1st Edition, Wiley Publications, 2013
- Dieter Uckelmann, Mark Harrison, Florian Michahelles, Architecting the Internet of Things, Springer-Verlag Berlin Heidelberg 2011 Industry 4.0: The Industrial Internet of Things

REFERENCES:

- Hakima Chaouchi, The Internet of Things Connecting Objects to the Web Willy Publications.
- Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things: Key Applications and Protocols, 2nd Edition, Wiley Publications
- Internet of Things - From Research and Innovation to Market Deployment; by Ovidiu Vermesan & Peter Friess; River Publishers Series,2014
- How Protocol Conversion Addresses IIoT Challenges: White Paper By RedLion.
- Alasdair Gilchrist, Industry 4.0: The Industrial Internet of Things, First edition, Kindle edition

24AM915	INTELLIGENT ROBOTS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the basics of Intelligent Robots.
- To discuss the Autonomous capabilities and Software architecture.
- To elaborate the Reactive Functionality of intelligent Robots.
- To use the various sensors in building Intelligent Robots.
- To illustrate the Deliberative Functionality of intelligent Robots.

UNIT I	INTRODUCTION	9

Overview- Definition – Components -Three Modalities – Need for Intelligent Robots – History of AI Robotics – Industrial Manipulators – Mobile Robots – Drones – Cognitive Systems.

UNIT II	AUTOMATION AND AUTONOMY	9

Autonomous Capabilities – Bounded Rationality – Automation and Autonomy – Programming Style – Hardware Design – Types of Functional Failures – Autonomous Capabilities. Types of Software Architectures – Operational Architectures – Components of a Telesystem – Human Supervisory Control.

UNIT III	REACTIVE FUNCTIONALITY	9

Behaviours : Agency and Marr's Computational Theory – Animal Behaviours – Schema Theory. Perception: Action-Perception cycle – Functions. Behaviour Coordination – Function – Cooperating Methods – Competing Methods – Sequences.

UNIT IV	SENSORS AND SENSING	9

Locomotion: Mechanical, Biomimetic, Legged Locomotion – Action Selection – Sensors and Sensing Model – Choosing – Range Sensing: Stereo – Depth from X – Sonar or Ultrasonics.

UNIT V	DELIBERATIVE FUNCTIONALITY	9

Deliberation – Strips – Navigation – Spatial Memory – Types of Path Planning – Configuration Space – Metric Path Planning – Motion Planning – Localization – Feature based Localization – Iconic Localization – Static vs Dynamic Environments – Simultaneous Localization and Mapping - Terrain Identification and Mapping – Scale and Traversability - Exploration – Mutlirobot Systems and AI – Human-Robot Interaction and areas of AI.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics of Intelligent Robots.
CO2: Design and implement Autonomous capabilities in Robotics systems.
CO3: Elaborate the Reactive Functionality of intelligent Robots.
CO4: Use the various sensors in building Intelligent Robots.
CO5: Illustrate the Deliberative Functionality of intelligent Robots.
CO6: Analyse the various applications of AI Robotics.

TEXT BOOKS:

1. Robin R. Murphy, "Introduction to AI Robotics", MIT Press, Second Edition, 2019.

REFERENCES:

1. Francis X. Govers, "Artificial Intelligence for Robotics: Build Intelligent Robots that Perform Human Tasks Using AI Techniques", Packt Publishing, 2018.
2. Sebastian Thrun, Wolfram Burgard, and Dieter Fox, "Probabilistic Robotics", MIT Press, 2005.
3. Yoon Seok Pyo, Han Cheol Cho, Ryu Woon Jung, and Tae Hoon Lim, "ROS Robot Programming", ROBOTIS Co., Ltd, 2017.

24AM916	CONVERSATIONAL AI	L 3	T 0	P 0	C 3

OBJECTIVES:

- Introduce fundamentals of Natural Language Processing (NLP) for dialogue systems
- Understand architectures of conversational agents
- Design and implement task-oriented and open-domain chatbots
- Apply machine learning and deep learning for dialogue understanding
- Evaluate, deploy, and improve conversational systems

UNIT I	FOUNDATIONS OF CONVERSATIONAL AI	9

Introduction to Conversational AI: history, evolution, applications - Types of conversational agents: Rule-based systems, Retrieval-based systems, Generative models- Basics of Natural Language Processing: Text preprocessing (tokenization, stemming, lemmatization), Syntax vs semantics - Introduction to dialogue systems: Turn-taking, context, intent, entities.

UNIT II	NATURAL LANGUAGE UNDERSTANDING (NLU)	9

Intent classification techniques: Rule-based vs ML-based - Entity recognition (NER) -Feature engineering: Bag-of-Words, TF-IDF - Introduction to embeddings: Word2Vec, GloVe - Classification models: Naïve Bayes, Logistic Regression, basic neural networks .

UNIT III	DIALOGUE MANAGEMENT & DESIGN	9

Dialogue system architecture: NLU → Dialogue Manager → NLG - Dialogue state tracking - Rule-based dialogue flow design - Reinforcement learning basics for dialogue policy - Conversational design principles: Context handling , Personalization , Error handling & fallback strategies.

UNIT IV	NATURAL LANGUAGE GENERATION & ADVANCED MODELS	9

Template-based vs generative response systems -Sequence-to-sequence models - Transformer-based models: Attention mechanism (conceptual) - Pretrained language models: BERT , GPT - Ethical issues: Bias, hallucination, safety .

UNIT V	DEPLOYMENT, EVALUATION & APPLICATIONS	9

Chatbot development platforms: Dialogflow , Rasa - Evaluation metrics: Accuracy, F1-score ,BLEU score (for responses) , User satisfaction metrics - Deployment: APIs, web integration - Applications: Healthcare, education, customer support .

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the fundamentals of conversational AI, NLP concepts, and types of dialogue systems.

CO2: Apply natural language understanding techniques to perform intent classification and entity recognition.

CO3: Design dialogue management strategies for building structured conversational systems.

CO4: Analyze advanced language generation models and their role in modern conversational AI systems.

CO5: Develop and evaluate a conversational AI application using appropriate tools and metrics.

TEXT BOOKS:

3. Speech and Language Processing, **Daniel Jurafsky & James H. Martin,** Pearson Education, **2nd Edition, 2009**
4. Natural Language Processing with Python, **Steven Bird, Ewan Klein, Edward Loper,** O'Reilly Media, **2009**

REFERENCES:

6. Designing Bots, **Amir Shevat,** O'Reilly Media, **2017**
7. Conversational AI, **Andrew Freed,** Manning Publications, **2021**
8. Deep Learning, **Ian Goodfellow, Yoshua Bengio, Aaron Courville,** MIT Press, **2016**
9. Building Chatbots with Python, **Sumit Raj**

24AM917	GAME DEVELOPMENT AND PROGRAMMING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To explain game programming fundamentals.
- To learn about the processes, mechanics, issues in game design.
- To gain knowledge of the game design and Artificial intelligence.
- To understand the design and scripting languages of game programming.
- To know about networked games and analyse code for sample games.

UNIT I	INTRODUCTION	9

Evolution of video game programming-The Game Loop-Time and games-Game objects-2D rendering Foundations-Sprites-Scrolling-Tile Maps-Vectors –Matrices

UNIT II	3D GRAPHICS FOR GAMES	9

Evolution of video game programming-The Game Loop-Time and games-Game objects-2D rendering Foundations-Sprites-Scrolling-Tile Maps-Vectors –Matrices

UNIT III	GAME DESIGN AND AI	9

Cameras-Types of cameras-Perspective projection-Camera implementation-Camera supportalgorithm- Real AI versus Game AI-Pathfinding-State based behaviours-Strategy and planning.

UNIT IV	USER INTERFACE AND SCRIPTING LANGUAGES	9

Menu system-HUD elements-Radar-other UI considerations-Scripting Languages-Implementing a scripting language-Tokenization-Syntax Analysis-Code Execution or Generation-Data Formats-Case study UI mods in world of warcraft.

UNIT V	NETWORKED GAMES	9

Protocols-Network Topology-Server/Client-Peer-to-Peer-Cheating-Sample game -Side scroller for iOS, Tower defense for PC/Mac-Code Analysis.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the essential 2D graphical and mathematical techniques for game programming.
CO2: Illustrate 3D graphics like coordinate spaces, lighting and shading, z-buffering, and quaternions
CO3: Apply artificial intelligence techniques in game design.
CO4: Construct a basic game engine using UI and scripting languages.
CO5: Develop code for sample games.

TEXT BOOK:

1. Sanjay Madhav, Game Programming Algorithms and Techniques:A platform -AgnosticApproach-Game Design,1st Edition, Addison-Wesley Professional,2013.

REFERENCES:

1. JouniSmed, HarriHakonen, Algorithms and Networking for Computer Games, 2nd Edition, Wiley Publications, 2017.
2. Ernest Adams and Andrew Rollings, “Fundamentals of Game Design”, Prentice Hall 3rd Edition, 2014. JungHyun Han, “3D Graphics for Game Programming”, Chapman and Hall/CRC, 1stEdition, 2011.

24AM918	AI FOR MARKETING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

1. Understand the challenges faced by modern marketers and the strategic role of AI in marketing.
2. Learn the essential concepts of networks, nodes, customer relationship moments, and AI/ML fundamentals for marketing.
3. Apply the AI Marketing Canvas framework across its five stages to develop AI-driven marketing strategies.
4. Analyze real-world implementations of AI in marketing across industries.
5. Evaluate the organizational change management and ethical implications of AI adoption in marketing.

UNIT I	THE CHALLENGE AND THE SOLUTION	9

The Marketer's Challenge Today – The AI Marketing Canvas: A Strategic Path Forward for Marketers – Navigating This Book – Networks and Nodes.

UNIT II	AI AND MARKETING ESSENTIALS	9

The Customer Relationship Moments Mental Model – What Are Artificial Intelligence and Machine Learning?

UNIT III	THE AI MARKETING CANVAS: FOUNDATION TO EXPANSION	9

Elements of the AI Marketing Canvas – Stage 1: Foundation – Stage 2: Experimentation – Stage 3: Expansion.

UNIT IV	THE AI MARKETING CANVAS: TRANSFORMATION TO MONETIZATION	9

Stage 4: Transformation – Stage 5: Monetization – Putting It All Together: Starbucks.

UNIT V	IMPLEMENTATION	9

Managing Change – Getting Started – A Call to Action.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Describe the challenges faced by modern marketers and explain the strategic role of the AI Marketing Canvas as a path forward. (K2)

CO2: Explain the concepts of networks, nodes, customer relationship moments, and the fundamentals of artificial intelligence and machine learning in the context of marketing. (K2)

CO3: Apply the five stages of the AI Marketing Canvas – Foundation, Experimentation, Expansion, Transformation, and Monetization – to develop AI-driven marketing strategies. (K3)

CO4: Analyze real-world AI marketing implementations such as the Starbucks case study to evaluate the effectiveness of AI strategies. (K4)

CO5: Design an AI marketing implementation plan incorporating change management, organizational readiness, and ethical considerations. (K5)

CO6: Evaluate the impact of AI adoption on marketing functions including advertising, sales, branding, pricing, and channel management. (K5)

TEXT BOOKS:

1. Venkatesan, R., and Lecinski, J., "The AI Marketing Canvas: A Five-Stage Road Map to Implementing Artificial Intelligence in Marketing," Stanford University Press, 2021.

REFERENCES:

1. Sterne, J., "Artificial Intelligence for Marketing: Practical Applications," John Wiley & Sons, 2017.
2. Gentsch, P., "AI in Marketing, Sales and Service: How Marketers without a Data Science Degree Can Use AI, Big Data and Bots," Springer, 2019.
3. King, K., "Using Artificial Intelligence in Marketing: How to Harness AI and Maintain the Competitive Edge," Kogan Page Publishers, 2019.
4. Hosanagar, K., "A Human's Guide to Machine Intelligence," Viking (Penguin), 2019.
5. Prof. Zillur Rahman, "AI in Marketing," NPTEL Online Course, IIT Roorkee, https://onlinecourses.nptel.ac.in/

AI AND CLOUD

24CS951	CLOUD FOUNDATIONS (Lab Integrated)	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To describe the different ways a user can interact with Cloud.
- To discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.
- To confer the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.
- To demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.
- To determine a variety of managed big data services in the cloud.

UNIT I	INTRODUCTION TO CLOUD	9

Cloud Computing - Cloud Versus Traditional Architecture - IaaS, PaaS, and SaaS - Cloud Architecture - The GCP Console - Understanding projects - Billing in GCP - Install and configure Cloud SDK - Use Cloud Shell - APIs - Cloud Console Mobile App.

UNIT II	COMPUTE AND STORAGE	9

Compute options in the cloud - Exploring IaaS with Compute Engine - Configuring elastic apps with autoscaling - Exploring PaaS - Event driven programs - Containerizing and orchestrating apps - Storage options in the cloud - Structured and unstructured storage in the cloud - Unstructured storage using Cloud Storage - SQL managed services - NoSQL managed services.

UNIT III	APIs AND SECURITY IN THE CLOUD	9

The purpose of APIs – API Services - Managed message services - Introduction to security in the cloud - The shared security model - Encryption options - Authentication and authorization with Cloud IAM - Identify Best Practices for Authorization using Cloud IAM.

UNIT IV	NETWORKING, AUTOMATION AND MANGAEMENT TOOLS	9

Introduction to networking in the cloud - Defining a Virtual Private Cloud - Public and private IP address basics - Cloud network architecture - Routes and firewall rules in the cloud - Multiple VPC networks - Building hybrid clouds using VPNs - Different options for load balancing - Introduction to Infrastructure as Code - Terraform - Monitoring and management tools.

UNIT V	BIG DATA AND MACHINE LEARNING SERVICES	9

Introduction to big data managed services in the cloud - Leverage big data operations - Build Extract, Transform, and Load pipelines - Enterprise Data Warehouse Services - Introduction to machine learning in the cloud - Building bespoke machine learning models with AI Platform - Pre-trained machine learning APIs.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Describe the different ways a user can interact with Cloud.

CO2: Discover the different compute options in Cloud and implement a variety of structured and unstructured storage models.

CO3: Discuss the different application managed service options in the cloud and outline how security in the cloud is administered in Cloud.

CO4: Demonstrate how to build secure networks in the cloud and identify cloud automation and management tools.

CO5: Discover a variety of managed big data services in the cloud.

CO6: Use Cloud services to build applications.

REFERENCES:

1. https://cloud.google.com/docs
2. https://www.cloudskillsboost.google/paths/36
3. https://nptel.ac.in/courses/106105223
4. Anthony J. Sequeira, "AWS Certified Cloud Practitioner (CLF-C01) Cert Guide", First Edition, Pearson Education, 2020.
5. AWS Documentation (amazon.com)
6. AWS Skill Builder
7. AWS Academy Cloud Foundations Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS501	DISTRIBUTED AND CLOUD COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To articulate the concepts and models underlying distributed computing
- To maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms.
- To learn different distributed mutual exclusion algorithms.
- To develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud.
- To explain the high-level automation and orchestration systems that manage the virtualized infrastructure.

UNIT I	INTRODUCTION	9

Definition - Relation to computer system components - Message-passing systems versus shared memory systems - Primitives for distributed communication - Synchronous versus asynchronous executions. A model of distributed computations: A distributed program - A model of distributed executions - Models of communication networks - Global state of a distributed system.

UNIT II	LOGICAL TIME, GLOBAL STATE, AND SNAPSHOT ALGORITHMS	9

Logical time – Scalar Time – Vector Time. Global state and snapshot recording algorithms: System model - Snapshot algorithms for FIFO channels and non-FIFO channels.

UNIT III	DISTRIBUTED MUTUAL EXCLUSION ALGORITHMS	9

Introduction - Lamport's algorithm - Ricart–Agrawala algorithm - Quorum-based mutual exclusion algorithms - Maekawa's algorithm - Suzuki–Kasami's broadcast algorithm.

UNIT IV	CLOUD INFRASTRUCTURE AND VIRTUALIZATION	9

Data Center Infrastructure and Equipment – Virtual Machines – Containers – Virtual Networks - Virtual Storage.

UNIT V	AUTOMATION AND ORCHESTRATION	9

Automation - Orchestration: Automated Replication and Parallelism - The MapReduce Paradigm: The MapReduce Programming Paradigm – Splitting Input – Parallelism and Data size – Data access and Data Transmission – Apache Hadoop – Parts of Hadoop – HDFS Components – Block Replication and Fault Tolerance – HDFS and MapReduce - Microservices.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Articulate the main concepts and models underlying distributed computing.

CO2: Learn how to maintain consistency and perform efficient coordination in distributed systems through the use of logical clocks, global states, and snapshot recording algorithms.

CO3: Learn different distributed mutual exclusion algorithms

CO4: Develop the ability to understand the cloud infrastructure and virtualization that help in the development of cloud.

CO5: Explain the high-level automation and orchestration systems that manage the virtualized infrastructure.

TEXT BOOKS:

1. Ajay D. Kshemkalyani, Mukesh Singhal, "Distributed Computing: Principles, Algorithms, and

Systems", Cambridge University Press, 2011. (Unit 1, 2, 3)
2. Douglass E. Comer, "The Cloud Computing Book: The future of computing explained", CRC Press, 2023. (Unit 4, 5)

REFERENCES:

1. Arshdeep Bahga, Vijay Madisetti, "Cloud Computing: A Hands-on Approach", Universities Press Private Limited, 2014.
2. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, "Mastering Cloud Computing", Tata Mcgraw Hill, 2024.
3. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2017.
4. Hagit Attiya, Jennifer Welch, "Distributed Computing: Fundamentals, Simulations and Advanced Topics", John Wiley & Sons, Inc., 2004.

LIST OF EQUIPMENTS:

Java, VMWare/VirtualBox, Hadoop, AWS/GCP/Azure/Any cloud platform, Eclipse/Equivalent IDE.

24CS912	VIRTUALIZATION	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Explain the fundamental concepts of virtualization
- Analyze the role of hypervisors in hardware virtualization
- Apply the understanding of CPU, memory (MMU), and I/O virtualization techniques
- Assess security considerations of virtualized environments
- Discuss strategies for protecting VMs and data centers

UNIT I	INTRODUCTION	9

Virtualization - Virtual Machines - Hypervisors - Type-1 and Type-2 Hypervisors - Multiplexing and Emulation - Approaches to Virtualization and Paravirtualization - Benefits of Using Virtual Machines. Working with Virtual Machines.

UNIT II	HARDWARE VIRTUALIZATION	9

The Popek/Goldberg Theorem - Virtualization without Architectural Support: Full Virtualization - Paravirtualization - Designs Options for Type-1 Hypervisors. Hypervisors: Describing a Hypervisor - Role of Hypervisor - VMWare ESX - Citrix Hypervisor - Microsoft Hyper-V.

UNIT III	TYPES OF VIRTUALIZATIONS	10

CPU Virtualization with VT-x: Design requirements - The VT-x Architecture - KVM. MMU Virtualization: Extended Paging - Virtualizing Memory in KVM. I/O Virtualization: Benefits of I/O Interposition - Physical I/O - Virtual I/O Without Hardware Support- Virtual I/O with Hardware Support. Virtualization Support in ARM Processors.

UNIT IV	VIRTUALIZATION SECURITY	9

Fundamentals of Virtualization Security: Virtualization Architecture - Threats to a Virtualized Environment. Securing Hypervisors: Hypervisor Configuration and Security. Designing Virtual Networks for Security: Comparing Virtual and Physical Networks - Virtual Network Security Considerations - Configuring Virtual Switches for Security.

UNIT V	VIRTUALIZATION AND AVAILABILITY	8

Availability - Protecting a Virtual Machine - Protecting Multiple Virtual Machines - Protecting Datacenters - Deploying Applications in a Virtual Environment - Recent Trends in Virtualization.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics of virtualization and its benefits.

CO2: Assess the significance of hypervisors in hardware virtualization, examining their roles and implications for system efficiency and performance

CO3: Utilize knowledge of virtualization technologies to solve practical problems and implement effective solutions

CO4: Analyze security threats and design secure virtual networks

CO5: Discuss strategies to improve availability in virtual environment and for protecting VMs and data centers

CO6: Use virtualization technology effectively to optimize system performance and resource usage in real-world settings

TEXTBOOKS:

1. Edouard Bugnion, Jason Nieh, Dan Tsafrir, “Hardware and Software Support for Virtualization”, Morgan & Claypool Publishers, 2017.
2. Matthew Portnoy, “Virtualization Essentials”, Third Edition, Sybex - John Wiley & Sons, 2023.

REFERENCES:

1. Dave Shackleford, “Virtualization Security: Protecting Virtualized Environments”, Sybex - John Wiley & Sons, 2012.
2. Nelson Ruest, Danielle Ruest, Virtualization, A beginners guide, 2009, McGrawHill.
3. Nadeau,Tim Cerng, Je Buller, Chuck Enstall, Richard Ruiz, Mastering Microsoft Virtualization,

Wiley Publication, 2010.
4. William Von Hagen, Professional Xen Virtualization, Wiley Publication, 2008.

24AM959	MLOps	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

- To understand the principles, challenges, and need for MLOps in modern AI systems.
- To learn the roles, responsibilities, and workflows involved in the MLOps lifecycle.
- To develop skills in model development, testing, validation, and reproducibility.
- To understand deployment, monitoring, governance, and maintenance of ML systems.
- To explore real-world MLOps applications across different domains

UNIT I	INTRODUCTION TO MLOPS AND ITS FOUNDATIONS	6+6

Need for MLOps – Defining MLOps, challenges in ML lifecycle management, risk assessment and mitigation, MLOps for Responsible AI, scalability considerations. People of MLOps – Subject Matter Experts, Data Scientists, Data Engineers, Software Engineers, DevOps Engineers, Model Risk Managers, Machine Learning Architects. Key Features of MLOps – Business objectives, data sources, exploratory data analysis, feature engineering, model training and evaluation, reproducibility, responsible AI principles, deployment overview, monitoring, governance, feedback loops, and lifecycle management.

List of Exercises/Experiments:

1. **Perform Exploratory Data Analysis (EDA)** on a standard dataset by identifying business objectives, exploring data sources, handling missing values, and applying basic feature engineering techniques.
2. **Build, train, and evaluate a Machine Learning model** using an appropriate dataset. Document the model performance using evaluation metrics and ensure reproducibility by saving the trained model.
3. **Deploy and monitor the trained Machine Learning model** using a simple deployment framework (Streamlit/Flask), record prediction logs, and demonstrate the concepts of monitoring, feedback loops, and lifecycle management in MLOps.

UNIT II	MODEL DEVELOPMENT AND VALIDATION	6+6

Machine Learning Models – theoretical and practical perspectives, required components of ML systems, challenges associated with different ML algorithms. Data Exploration and Feature Engineering – feature creation, selection techniques, impact on MLOps strategy. Experimentation and Evaluation – model comparison, evaluation metrics, cross-validation, model behavior analysis. Responsible AI in Modeling – fairness, transparency, explainability. Version Management and Reproducibility – experiment tracking, model versioning, reproducible workflows.

List of Exercises/Experiments:

1. **Build and compare Machine Learning models** (e.g., Decision Tree, Random Forest, and Logistic Regression) using a standard dataset, and analyze their performance using appropriate evaluation metrics.
2. **Perform feature engineering and model evaluation** by applying feature selection techniques, cross-validation, and analyzing the impact of different features on model performance.
3. **Implement experiment tracking and model versioning** by saving different model versions, documenting experiments, and evaluating fairness, transparency, and reproducibility in the machine learning workflow.

UNIT III	PRODUCTION READINESS AND DEPLOYMENT	6+6

Preparing Models for Production – runtime environments, migration from development to production environments, data access considerations. Model Risk Evaluation – validation processes, quality assurance, testing methodologies, auditability and reproducibility. Machine Learning Security – adversarial attacks, vulnerabilities, risk mitigation strategies. Deployment to Production – CI/CD pipelines, ML artifacts, testing pipelines, deployment strategies, deployment categories, production considerations, containerization, scalability challenges, maintenance of deployed models.

List of Exercises/Experiments:

1. **Prepare and deploy a Machine Learning model** by creating a runtime environment,

packaging the trained model, and deploying it using **Streamlit/Flask** or a **Docker** container.
2. **Implement model validation and testing** by performing quality assurance, testing model performance with different test datasets, and maintaining model versions to ensure auditability and reproducibility.
3. **Design a basic CI/CD pipeline for Machine Learning deployment** by integrating model artifacts, automating deployment, monitoring model performance, and analyzing security and scalability considerations in a production environment.

UNIT IV	MONITORING, FEEDBACK, AND GOVERNANCE	6+6

Monitoring Machine Learning Systems – model degradation, retraining strategies, ground truth evaluation, input drift detection, drift detection techniques, logging and performance monitoring. Feedback Loop Mechanisms – online evaluation, continuous improvement,retraining workflows. Model Governance – governance frameworks, risk-based governance, regulatory requirements, GDPR, CCPA, AI regulations. Responsible AI Governance – data governance, bias mitigation, inclusiveness, governance policies, centralized governance management, monitoring and refinement of governance processes.

List of Exercises/Experiments:

1. **Monitor the performance of a deployed Machine Learning model** by detecting model degradation, analyzing input data drift, and evaluating prediction accuracy using appropriate monitoring metrics.
2. **Implement a feedback loop for continuous model improvement** by collecting prediction data, evaluating model performance with ground truth, and performing model retraining.
3. **Analyze governance and Responsible AI practices** by documenting data governance policies, identifying bias in model predictions, and evaluating compliance with AI governance principles and regulatory requirements (e.g., GDPR and CCPA).

UNIT V	MLOPS CASE STUDIES AND INDUSTRIAL APPLICATIONS	6+6

Consumer Credit Risk Management – business use cases, model development, bias considerations, deployment and monitoring. Marketing Recommendation Engines – experimentation, model training, deployment strategies, scalability, real-time scoring, monitoring and retraining. Consumption Forecasting Systems – data collection, problem definition, modeling, deployment, monitoring. Comparative analysis of MLOps practices across domains and emerging industry trends

List of Exercises/Experiments:

1. **Develop and evaluate a Consumer Credit Risk prediction model** using a standard dataset, analyze model performance, identify bias, and discuss deployment and monitoring strategies.
2. **Build and deploy a Recommendation System** using a suitable dataset, perform model training and evaluation, and analyze scalability, real-time prediction, and retraining requirements.
3. **Develop a Consumption Forecasting model** using historical data, deploy the model, monitor its performance, and compare MLOps practices across credit risk, recommendation, and forecasting applications.

TOTAL: 30+30=60 PERIODS

OUTCOMES:

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

CO1: Explain the concepts, objectives, challenges, and key stakeholders involved in the MLOps lifecycle.

CO2: Apply data exploration, feature engineering, experimentation, and model evaluation techniques to develop machine learning solutions.

CO3: Implement reproducible machine learning workflows using version control, experiment tracking, and validation practices.

CO4: Design and deploy production-ready machine learning pipelines using CI/CD, containerization, testing, and scalability techniques.

CO5: Analyze and monitor deployed machine learning models through drift detection, performance evaluation, feedback loops, and retraining

strategies.

CO6: Evaluate governance, security, ethical AI, and regulatory requirements while applying MLOps practices to real-world industrial use cases.

TEXT BOOKS:

1. Introducing MLOps, How to Scale Machine Learning in the Enterprise, Mark Treveil and the Dataiku Team, O'Reilly, 2020.

REFERENCES:

1. Practical MLOps: Operationalizing Machine Learning Models, Noah Gift and Alfredo Deza, O'Reilly Media, 2021 .
2. Machine Learning Engineering, Andriy Burkov, True Positive Inc., 2020 .

24CS913	DATA ENGINEERING IN CLOUD	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- ✓ Grasp the fundamentals of data engineering, emphasizing cloud-based data access.
- ✓ Construct robust and secure data pipelines using Cloud services.
- ✓ Select and implement appropriate data storage solutions while prioritizing pipeline security.
- ✓ Utilize cloud tools for handling extensive data for machine learning purposes.
- ✓ Efficiently analyze, visualize, and automate data pipelines to streamline operations.

UNIT I	INTRODUCTION	8

Introduction to data Engineering - The Data Engineering Life Cycle - Data Engineering and Data Science - Data-Driven Organizations: Data-driven decisions - The data pipeline - The role of the data engineer in data-driven organizations - Modern data strategies - The Elements of Data: The five Vs of data – volume, velocity, variety, veracity, and value. Demo: Accessing and Analyzing Data by Using Amazon S3.

UNIT II	SECURE AND SCALABLE DATA PIPELINES	10

The evolution of data architectures - Modern data architecture on AWS - Modern data architecture pipeline: Ingestion and storage - Processing and consumption - Streaming analytics pipeline - Security of analytics workloads - Scaling - Creating a scalable infrastructure and components. ETL and ELT comparison - Data wrangling.

UNIT III	STORING AND ORGANIZING DATA	9

Comparing batch and stream ingestion - Batch ingestion processing - Purpose-built ingestion tools - AWS Glue for batch ingestion processing - Kinesis for stream processing - Scaling considerations for batch processing and stream processing - Storage in the modern data architecture - Data lake storage - Data warehouse storage - Purpose-built databases - Storage in support of the pipeline - Securing storage.

UNIT IV	PROCESSING BIG DATA AND DATA FOR ML	10

Big data processing concepts - Apache Hadoop - Apache Spark - Amazon EMR - Managing your Amazon EMR clusters - Apache Hudi - The ML lifecycle - Collecting data - Applying labels to training data with known targets - Preprocessing data - Feature engineering - Developing a model - Deploying a model - ML infrastructure on AWS - SageMaker - Amazon CodeWhisperer - AI/ML services on AWS.

UNIT V	DATA ANALYSIS AND VISUALIZATION	8

Analyzing and Visualizing Data: Considering factors that influence tool selection - Comparing AWS tools and services - Selecting tools for a gaming analytics use case. Automating the Pipeline: Automating infrastructure deployment - CI/CD - Automating with Step Functions.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand data engineering, pipelines & access data in the cloud.
CO2: Build secure & scalable data pipelines using AWS services.
CO3: Choose the right data storage & secure your data pipelines.
CO4: Process big data for machine learning with cloud tools.
CO5: Analyze & visualize data and automate data pipelines.
CO6: Apply best practices in data governance, compliance, and ethics throughout the data engineering process, ensuring responsible handling and usage of data.

TEXT BOOKS:

2. Martin Kleppman, "Data Engineering: Building Reliable Scalable Data Systems", O'Reilly Media, 2017.
3. Wes McKinney, "Python for Data Analysis", 2nd Edition, O'Reilly Media, 2017.

REFERENCES:

1. Martin Kleppman, "Designing Data-Intensive Applications", O'Reilly Media, 2017.

2. AWS Documentation (amazon.com)
3. AWS Skill Builder
4. AWS Academy Data Engineering Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS920	CLOUD SERVICES MANAGEMENT	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- ✓ Introduce Cloud Service Management terminology, definition & concepts
- ✓ Compare and contrast cloud service management with traditional IT service management
- ✓ Identify strategies to reduce risk and eliminate issues associated with adoption of cloud services
- ✓ Select appropriate structures for designing, deploying and running cloud-based services in a business environment
- ✓ Illustrate the benefits and drive the adoption of cloud-based services to solve real world problems

UNIT I	CLOUD SERVICE MANAGEMENT FUNDAMENTALS	9

Cloud Ecosystem, The Essential Characteristics, Basics of Information Technology Service Management and Cloud Service Management, Service Perspectives, Cloud Service Models, Cloud Service Deployment Models.

UNIT II	CLOUD SERVICES STRATEGY	9

Cloud Strategy Fundamentals, Cloud Strategy Management Framework, Cloud Policy, Key Driver for Adoption, Risk Management, IT Capacity and Utilization, Demand and Capacity matching, Demand Queueing, Change Management, Cloud Service Architecture.

UNIT III	CLOUD SERVICE MANAGEMENT	9

Cloud Service Reference Model, Cloud Service LifeCycle, Basics of Cloud Service Design, Dealing with Legacy Systems and Services, Benchmarking of Cloud Services, Cloud Service Capacity Planning, Cloud Service Deployment and Migration, Cloud Marketplace, Cloud Service Operations Management.

UNIT IV	CLOUD SERVICE ECONOMICS	9

Pricing models for Cloud Services, Freemium, Pay Per Reservation, Pay per User, Subscription based Charging, Procurement of Cloud-based Services, Capex vs Opex Shift, Cloud service Charging, Cloud Cost Models.

UNIT V	CLOUD SERVICE GOVERNANCE & VALUE	9

IT Governance Definition, Cloud Governance Definition, Cloud Governance Framework, Cloud Governance Structure, Cloud Governance Considerations, Cloud Service Model Risk Matrix, Understanding Value of Cloud Services, Measuring the value of Cloud Services, Balanced Scorecard, Total Cost of Ownership.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Exhibit cloud-design skills to build and automate business solutions using cloud technologies.

CO2: Possess Strong theoretical foundation leading to excellence and excitement towards adoption of cloud-based services

CO3: Solve the real world problems using Cloud services and technologies

CO4: Develop and deploy services on the cloud and set up a cloud environment

CO5: Explain security challenges in the cloud environment

CO6: Demonstrate proficiency in integrating cloud technologies and services to address diverse business challenges effectively.

TEXT BOOKS:

1. Enamul Haque, "Cloud Service Management and Governance: Smart Service Management in Cloud Era", Enel Publications, 2023.
2. Thomas Erl, Ricardo Puttini, Zaigham Mohammad, "Cloud Computing: Concepts, Technology & Architecture", Prentice Hall, 2013.

REFERENCES:

1. Thomas Erl, Robert Cope, Amin Naserpour, "Cloud Computing Design Patterns", Prentice

Hall, 2015.
2. Praveen Ayyappa, "Economics of Cloud Computing", LAP Lambert Academic Publishing, 2020.
3. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "Mastering Cloud Computing Foundations and Applications Programming", Elsevier, 2013.

24AM945	FRAMEWORKS FOR CLOUD DATA ANALYTICS	L	T	P	C
		2	0	2	3

OBJECTIVES: The Course will enable learners to:

- Understand the architecture and key components of the Snowflake Data Cloud and the Databricks Lakehouse Platform.
- Learn to create and manage databases, schemas, tables, and other securable objects in Snowflake.
- Develop skills in data loading, unloading, transformation, and query optimization using SQL on cloud platforms.
- Explore Delta Lake for managing data with ACID transactions and use Apache Spark for data transformation on Databricks.
- Build production data pipelines, implement data governance, and perform analytics using Databricks SQL.
- Apply best practices for access control, cost management, performance tuning, and data sharing on cloud data platforms.

UNIT I	GETTING STARTED WITH SNOWFLAKE AND DATABRICKS (Snowflake: Ch 1–2, Databricks: Ch 1)	6+6

Getting Started with Snowflake: Snowflake Web User Interfaces – Snowsight Orientation – Snowsight Preferences – Navigating Snowsight Worksheets – Context Setting.
Creating and Managing the Snowflake Architecture: Snowflake Architecture Overview – Cloud Services Layer – Query Processing (Virtual Warehouses) – Database Storage – Creating and Managing Virtual Warehouses – Resource Monitors.
Getting Started with Databricks: Introducing the Databricks Platform – Understanding the Databricks Platform – High-Level Architecture of the Databricks Lakehouse – Apache Spark on Databricks – Databricks File System (DBFS) – Setting Up a Databricks Workspace – Exploring the Databricks Workspace – Creating Clusters – Working with Notebooks.

List of Exercises/Experiments:

1. Set up a Snowflake trial account, explore the Snowsight interface, create a virtual warehouse, and execute basic SQL queries on sample data.
2. Set up a Databricks Community Edition workspace, create a cluster, create a notebook, and run basic PySpark commands to read and display a CSV dataset.
3. Compare the architectures of Snowflake and Databricks by documenting key differences in compute, storage, and interface features using the trial accounts.

UNIT II	DATABASE OBJECTS, SQL, AND DATA MANAGEMENT (Snowflake: Ch 3–6)	6+6

Creating and Managing Snowflake Securable Database Objects: Database and Schema Management – Permanent, Transient, and Temporary Tables – Views: Standard, Secure, and Materialized – User-Defined Functions (UDFs) – Stored Procedures – Sequences – Snowflake Streams – Snowflake Tasks.
Exploring Snowflake SQL Commands, Data Types, and Functions: Data Definition Language (DDL) Commands – Data Manipulation Language (DML) Commands – Snowflake Data Types – Numeric, String, Date/Time, and Semi-Structured Data Functions – Working with JSON, Parquet, and XML Data.
Leveraging Snowflake Access Controls: Role-Based Access Control (RBAC) – Discretionary Access Control (DAC) – System-Defined Roles – Custom Roles – Granting and Revoking Privileges.
Data Loading and Unloading: Basics of Data Loading and Unloading – File Formats – Snowflake Stage References – COPY INTO Commands – Snowpipe for Continuous Data Loading – Data Loading Best Practices.

List of Exercises/Experiments:

1. Create databases, schemas, tables (permanent, transient, temporary), views, and sequences in Snowflake. Insert sample data and query using standard SQL commands.

2. Load a CSV file into Snowflake using internal stages and the COPY INTO command. Query semi-structured JSON data using Snowflake's VARIANT data type and lateral flatten.
3. Configure role-based access control by creating custom roles, granting privileges on database objects, and verifying access restrictions across different roles.

UNIT III	DELTA LAKE AND DATA TRANSFORMATION ON DATABRICKS (Databricks: Ch 2–4)	6+6

Managing Data with Delta Lake: What is Delta Lake – Creating Delta Tables – Inserting Data into Delta Tables – Updating and Deleting Data – Upsert Using MERGE INTO – Table History and Versioning – Time Travel – Optimizing Delta Tables: OPTIMIZE and ZORDER – Vacuum.
Mastering Relational Entities in Databricks: Databases and Schemas – Tables: Managed and External – Views: Standard, Temporary, and Global Temporary Views – Common Table Expressions (CTEs).
Transforming Data with Apache Spark: Querying Files Directly – Providing Schema for External Data Sources – Creating Tables from External Sources – Writing to Tables – Cleaning Data – Complex Transformations – Working with JSON and Nested Data – Spark SQL Built-In Functions.

List of Exercises/Experiments:

1. Create a Delta table in Databricks, perform INSERT, UPDATE, DELETE, and MERGE operations. Use Time Travel to query previous versions and restore data.
2. Load a JSON dataset into Databricks, flatten nested structures using Spark SQL, create managed and external tables, and apply data cleaning transformations.
3. Run OPTIMIZE and ZORDER on a Delta table with at least 100,000 rows and compare query performance before and after optimization using EXPLAIN.

UNIT IV	PRODUCTION PIPELINES, GOVERNANCE, AND ANALYTICS (Databricks: Ch 5–8, Snowflake: Ch 7)	6+6

Processing Incremental Data: Structured Streaming – Streaming with Delta Lake – Auto Loader – Multi-Hop (Medallion) Architecture: Bronze, Silver, and Gold Layers.
Building Production Pipelines: Delta Live Tables (DLT) – DLT Expectations for Data Quality – Configuring DLT Pipelines – Capturing Data Changes (CDC) – Orchestrating Workflows – Databricks Jobs.
Exploring Databricks SQL: Creating SQL Warehouses – Designing Dashboards – Managing SQL Queries – Setting Up Alerts.
Implementing Data Governance: Unity Catalog – Data Access Control – Managing Permissions – Data Lineage.
Snowflake Data Sharing and Collaboration: Secure Data Sharing – Snowflake Marketplace – Data Exchange – Reader Accounts.

List of Exercises/Experiments:

1. Build a multi-hop (medallion) data pipeline in Databricks using Structured Streaming with Auto Loader to ingest raw CSV files into Bronze, transform into Silver, and aggregate into Gold Delta tables.
2. Create a Databricks SQL warehouse, write analytical queries on the Gold layer tables, and build an interactive dashboard with visualizations and scheduled alerts.
3. Configure Secure Data Sharing in Snowflake by creating a share, adding database objects, and simulating consumer access using a reader account.

UNIT V	PERFORMANCE OPTIMIZATION, COST MANAGEMENT, AND INTEGRATED ANALYTICS (Snowflake: Ch 8–12)	6+6

Managing Snowflake Account Costs: Snowflake Credit Usage – Monitoring Storage and Compute Costs – Resource Monitors – Cost Optimization Strategies.
Analyzing and Improving Snowflake Query Performance: Query Profile and Query History – Caching: Result Cache, Metadata Cache, Warehouse Cache – Clustering Keys – Search Optimization Service – Materialized Views for Performance – Query Optimization Best Practices.
Continuous Data Protection and Data Sharing: Time Travel – Fail-Safe – Zero-Copy Cloning –

Data Replication.

Workloads for the Snowflake Data Cloud: Data Engineering – Data Warehousing – Data Lake – Data Analytics – Data Science – Snowpark – Streamlit.

List of Exercises/Experiments:

1. Analyze query performance in Snowflake using the Query Profile. Apply clustering keys on a large table and measure improvement in query execution time.
2. Implement Snowflake Time Travel and Zero-Copy Cloning: drop and restore a table using Time Travel, and create a zero-copy clone for development and testing.
3. Mini-Project: Design and implement an end-to-end cloud analytics solution that ingests raw data into Databricks (Bronze–Silver–Gold pipeline), loads the Gold layer into Snowflake using an external stage, and creates a dashboard for business analytics.

TOTAL: 30+30 = 60 PERIODS

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

CO1: Explain the architecture, components, and workloads of the Snowflake Data Cloud and the Databricks Lakehouse Platform. *[K2 – Understand]*

CO2: Create and manage databases, schemas, tables, views, and access controls in Snowflake using SQL. *[K3 – Apply]*

CO3: Apply Delta Lake operations and Apache Spark transformations for data management and processing on Databricks. *[K3 – Apply]*

CO4: Analyze query performance and optimize data processing using caching, clustering keys, and best practices on Snowflake. *[K4 – Analyze]*

CO5: Design end-to-end data pipelines using medallion architecture, Delta Live Tables, and orchestration workflows on Databricks. *[K6 – Create]*

CO6: Evaluate data governance, cost management, and secure data sharing strategies across Snowflake and Databricks platforms. *[K5 – Evaluate]*

TEXT BOOKS:

1. Joyce Kay Avila, "Snowflake: The Definitive Guide: Architecting, Designing, and Deploying on the Snowflake Data Cloud", O'Reilly Media, 2022. ISBN: 9781098103828.
2. Derar Alhussein, "Databricks Certified Data Engineer Associate Study Guide", O'Reilly Media, 2025. ISBN: 9781098166816.

REFERENCES:

1. Bill Chambers and Matei Zaharia, "Spark: The Definitive Guide: Big Data Processing Made Simple", O'Reilly Media, 2018. ISBN: 9781491912218.
2. Denny Lee, Tristen Wentling, Scott Haines, and Prashanth Babu, "Delta Lake: The Definitive Guide: Modern Data Lakehouse Architectures with Data Lakes", O'Reilly Media, 2024. ISBN: 9781098151942.
3. Snowflake Official Documentation. Freely available at https://docs.snowflake.com
4. Databricks Official Documentation. Freely available at https://docs.databricks.com
5. Ralph Kimball and Margy Ross, "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling", 3rd Edition, Wiley, 2013. ISBN: 9781118530801.

24CS915	MACHINE LEARNING FOR NLP IN CLOUD	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Illustrate how to apply the ML pipeline to NLP.
- Implement text extraction to obtain data from web pages.
- Build a solution that uses AWS services to transcribe and translate text from multimedia.
- Build a solution using a combination of algorithms and Amazon Machine Learning (Amazon ML) services.
- Identify use cases to use generative AI and LLMs.
- Use LLMs with AWS generative AI services.

UNIT I	INTRODUCTION TO NLP	8

NLP – Business Problems Solved by NLP – NLP Roles - NLP and ML – Common NLP tasks – Apply ML to NLP problem - Evolution of NLP architectures.

UNIT II	PROCESSING TEXT FOR NLP	10

Text processing overview - Getting text - Extracting Text from Webpages and Images - Text preprocessing - Vectorizing text - Encoding and Vectorizing Text - Advanced processing - Storing and visualizing unstructured data – Implement Sentiment Analysis: Identifying the steps for text processing - Examining the algorithms for sentiment analysis.

UNIT III	INFORMATION EXTRACTION	9

Information extraction overview - Types of information extraction - Implementing information extraction – Working with Entities - Topic Modeling - Identifying the approach - Implementing Topic Modeling with Amazon Comprehend, Neural Topic Model (NTM).

UNIT IV	TRANSLATING LANGUAGES	9

Working with language issues - Detecting and translating languages - Transcribing and vocalizing text with AWS services - Implementing a Multilingual Solution.

UNIT V	GENERATIVE AI	9

Generative AI - Amazon Bedrock Overview - Introducing foundations models and large language models - Transformer architecture - LLMs configuration parameters - Introducing prompt engineering - Use LLMs to Perform NLP Tasks - Adapting LLMs - Application Integration.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand basic concepts and applications of NLP.
CO2: Apply text preprocessing and vectorization techniques.
CO3: Analyze and perform sentiment analysis on text data.
CO4: Apply information extraction and topic modeling methods.
CO5: Implement language translation and speech-based NLP tasks.
CO6: Use Generative AI and LLMs for NLP applications.

REFERENCES:

1. Mona M, Premkumar Rangarajan, Natural Language Processing with AWS AI Services, Packt Publications, 2021.
2. Saket S R Mengle, Maximo Gurmendez, Mastering Machine Learning on AWS: Advanced machine learning in Python using SageMaker, Apache Spark, and TensorFlow, Packt Publications, 2019.
3. AWS Documentation (amazon.com)
4. AWS Skill Builder
5. AWS Academy Machine Learning for Natural Language Processing Course - https://www.awsacademy.com/vforcesite/LMS_Login

24CS916	MICROSERVICE ARCHITECTURE	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Learn the Concepts of Micro Service Architecture
- Understand the inter process communication in a microservice architecture
- Design business logic in a microservice architecture
- Develop business logic with event sourcing
- Implement queries in a microservice architecture

UNIT I	INTRODUCTION TO MICROSERVICE ARCHITECTURE	9

Introduction - Microservice - Microservice Architecture - Comparing Microservice Architecture with SOA - Benefits and Drawbacks of Microservice Architecture - Microservice Architecture Pattern Language - Beyond microservices -Process and Organization - Microservice Architecture Style - Application Microservice Architecture.

UNIT II	INTERPROCESS COMMUNICATION AND TRANSACTION MANAGEMENT	9

Overview of Inter process Communication - Synchronous Remote Procedure Invocation Pattern - Asynchronous Messaging Pattern - Asynchronous Messaging to Improve Availability - Transaction Management in a Microservice Architecture - Coordinating sagas - Handling the Lock of Isolation - Order Service and Order Saga.

UNIT III	DESIGN AND DEVELOPMENT OF BUSINESS LOGIC	9

Business logic organization patterns - Designing a domain model using the DDD aggregate pattern - Publishing domain events - Kitchen Service business logic - Order Service business logic - Developing business logic with event sourcing - Implementing an event store - Using sagas and event sourcing together.

UNIT IV	QUERY IMPLEMENTATION, EXTERNAL API PATTERNS AND MICROSERVICE TESTING	9

Querying using the API composition pattern - Using the CQRS pattern - Designing CQRS views - Implementing a CQRS view with AWS DynamoDB - External API design issues - The API gateway pattern - Implementing an API gateway - Testing strategies for microservice architectures - Writing unit tests for a service - Writing integration tests - Developing component tests - Writing end-to-end tests.

UNIT V	MICROSERVICES DEVELOPMENT AND DEPLOYMENT	9

Developing secure services - Designing configurable services - Designing observable services - Developing services using the Microservice chassis pattern - Deploying Microservices - Language-specific packaging format pattern - Service as a virtual machine pattern - Service as a container pattern - Serverless deployment - Refactoring to microservices.

TOTAL : 45 PERIODS

OUTCOMES:

Upon completion of the course, the students will be able to: CO1:
Develop microservice applications.

CO2: Deploy microservices in different platforms

CO3: Create integration and component testing for microservices

CO4: Design External API patterns for microservices

CO5: Develop inter process communication mechanisms in microservice architecture

CO6: Create transaction management in microservice.

TEXT BOOKS:

1. Chris Richardson, "Microservices Pattern with Examples in JAVA", Manning Publications, 1st edition, 2019.

REFERENCES:

1. Irakli Nadareishvili, Ronnie Mitra, Matt McLarty & Mike Amundsen, "Microservice Architecture ALIGNING PRINCIPLES, PRACTICES, AND CULTURE," 1st edition, O'Reilly Publication, 2016.
2. Eberhard Wolff, "Microservices Flexible Software Architectures," 1st edition, Addison- Wesley, 2016.
3. John Kernal, IIIary Huaylupo Sanchez, "Micro Service in Action", 2nd edition, Manning Publication,2018.
4. Susan J Fowler, "Production-Ready Microservices", O'Reilly Publication, 2016.
5. Sam Newman, "Building Microservices: Designing Fine-Grained Systems", 2nd edition, O'Reilly Publication, 2021.

HIGH PERFORMANCE COMPUTING

24AM919	MULTI-CORE ARCHITECTURES AND PROGRAMMING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the need for multi-core processors, and their architecture.
- To understand the challenges in parallel and multi-threaded programming.
- To learn about the various parallel programming paradigms.
- To develop multi core programs.
- To design parallel solutions.

UNIT I	MULTI-CORE PROCESSORS	9

Single core to Multi-core architectures – SIMD and MIMD systems – Interconnection networks - Symmetric and Distributed Shared Memory Architectures – Cache coherence - Performance Issues – Parallel program design

UNIT II	PARALLEL PROGRAM CHALLENGES	9

Performance – Scalability – Synchronization and data sharing – Data races – Synchronization primitives (mutexes, locks, semaphores, barriers) – deadlocks and livelocks – communication between threads (condition variables, signals, message queues and pipes).

UNIT III	SHARED MEMORY PROGRAMMING WITH OpenMP	9

Compiling and running OpenMP programs,The Trapezoidal rule,The parallel for directive, scheduling loops- Producers and consumers .

UNIT IV	DISTRIBUTED MEMORY PROGRAMMING WITH MPI	9

MPI program execution – MPI constructs – libraries – MPI send and receive – Point-to-point and Collective communication – MPI derived datatypes – Performance evaluation.

UNIT V	PARALLEL PROGRAM DEVELOPMENT	9

Case studies - n-Body solvers – Tree Search – OpenMP and MPI implementations and comparison.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Illustrate multicore architectures and identify their characteristics and challenges.
CO2: Identify the issues in programming Parallel Processors.
CO3: Write programs using OpenMP and MPI.
CO4: Design parallel programming solutions to common problems.
CO5: Compare and contrast programming for serial processors and programming for parallel processors.
CO6: Elaborate on various concepts of multi-core architectures.

TEXT BOOKS:

1. Peter S. Pacheco, “An Introduction to Parallel Programming”, Morgan-Kauffman/Elsevier, 2011.
2. Darryl Gove, “Multicore Application Programming for Windows, Linux, and Oracle Solaris”, Pearson, 2011.

REFERENCES:

1. Michael J Quinn, “Parallel programming in C with MPI and OpenMP‖”, Tata McGraw Hill,2003.
2. Victor Alessandrini, “Shared Memory Application Programming Concepts and Strategies in Multicore Application Programming, ”, 1st Edition, Morgan Kaufmann, 2015.
3. Yan Solihin, “Fundamentals of Parallel Multicore Architecture”, CRC Press, 2015.

24AM920	GPU ARCHITECTURES AND PROGRAMMING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the basics of GPU architectures
- To write programs for massively parallel processors
- To understand the issues in mapping algorithms for GPUs
- To introduce different GPU programming models

UNIT I	GPU ARCHITECTURE	9

Evolution of GPU architectures – Understanding Parallelism with GPU –Typical GPU Architecture – CUDA Hardware Overview – Threads, Blocks, Grids, Warps, Scheduling – Memory Handling with CUDA: Shared Memory, Global Memory, Constant Memory and Texture Memory.

UNIT II	CUDA PROGRAMMING	9

Using CUDA – Multi GPU – Multi GPU Solutions – Optimizing CUDA Applications: Problem Decomposition, Memory Considerations, Transfers, Thread Usage, Resource Contentions.

UNIT III	PROGRAMMING ISSUES	9

Common Problems: CUDA Error Handling, Parallel Programming Issues, Synchronization, Algorithmic Issues, Finding and Avoiding Errors.

UNIT IV	OPENCL BASICS	9

OpenCL Standard – Kernels – Host Device Interaction – Execution Environment – Memory Model – Basic OpenCL Examples.

UNIT V	ALGORITHMS ON GPU	9

Parallel Patterns: Convolution, Prefix Sum, Sparse Matrix – Matrix Multiplication – Programming Heterogeneous Cluster.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Able to describe GPU Architecture

CO2: Write programs using CUDA, identify issues and debug them

CO3: Implement efficient algorithms in GPUs for common application kernels, such as matrix multiplication

CO4: Write simple programs using OpenCL

CO5: Identify efficient parallel programming patterns to solve problems

CO6: Explore techniques for multi-GPU programming and distributed parallelism.

TEXT BOOKS:

1. Shane Cook, CUDA Programming: A Developers Guide to Parallel Computing with GPUs (Applications of GPU Computing), First Edition, Morgan Kaufmann, 2012.(Unit – I,II,III)
2. Benedict Gaster,Lee Howes, David R. Kaeli , “Heterogeneous Computing with OpenCL” Published by Elsevier, 2012.(Unit – IV,V)

REFERENCES:

1. David R. Kaeli, Perhaad Mistry, Dana Schaa, Dong Ping Zhang, “Heterogeneous computing with OpenCL”, 3rd Edition, Morgan Kauffman, 2015.
2. David B. Kirk, Wen-mei W. Hwu, “Programming Massively Parallel Processors – A Hands-on Approach”, Third Edition, Morgan Kaufmann, 2016.
3. David Kirk and Wen-mei Hwu, "Programming Massively Parallel Processors" – 4th Edition, Morgan Kaufmann, 2022
4. http://www.nvidia.com/object/cuda_home_new.html
5. http://www.openCL.org
6. Nicholas Wilt, CUDA Handbook: A Comprehensive Guide to GPU Programming, Addison - Wesley, 2013.
7. https://onlinecourses.nptel.ac.in/noc25_cs37/preview?

24AM921	HIGH PERFORMANCE COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the basics of Modern processors.
- To analyze the various optimization techniques for writing parallel high-performance applications.
- To discuss the basics of Parallel computers.
- To learn shared memory parallel programming using OpenMP.
- To understand the distributed memory parallel programming with MPI.

UNIT I	MODERN PROCESSORS	9

Stored Program Computer Architecture - General purpose cache- based microprocessor-Performance based metrics and benchmarks- Moore's Law- Pipelining- Superscalarity - SIMD- Memory Hierarchies - Cache- mapping- prefetch- Multicore processors- Mutithreaded processors- Vector Processors- Design Principles- Maximum performance estimates- Programming for vector architecture.

UNIT II	OPTIMIZATION TECHNIQUES	9

Basic optimization techniques for serial code : scalar profiling - function and line based runtime profiling- hardware performance counters- common sense optimizations- elimination of common subexpressions- avoiding branches - using SIMD instruction sets- the role of compilers - C++ optimizations - data access optimization: balance analysis and light speed estimates- storage order- Case study: jacobi algorithm and dense matrix transpose.

UNIT III	PARALLEL COMPUTERS	9

Taxonomy of parallel computing paradigms - Shared memory computers - Distributed-memory computers- Hierarchical systems- Networks - Basics of parallelization – Need to parallelize - Parallelism - Parallel Scalability- Factors that limit parallel execution- Scalability metrics- Simple scalability laws- parallel efficiency - serial performance Vs Strong scalability- Load balance.

UNIT IV	SHARED MEMORY PARALLEL PROGRAMMING WITH OpenMp	9

Introduction to Open MP - parallel execution - data scoping- OpenMp work sharing for loops- synchronization - reductions - loop scheduling - tasking - case study: OpenMp-parallel jacobi algorithm - Efficient OpenMP programming: Profiling OpenMP programs - Performance pitfalls.

UNIT V	DISTRIBUTED MEMORY PARALLEL PROGRAMMING WITH MPI	9

Message passing - introduction to MPI – example - messages and point-topoint communication - collective communication – nonblocking point-to-point communication- virtual topologies - MPI parallelization of Jacobi solver- MPI implementation - performance properties - Basic MPI/OpenMP programming models.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics of Modern processors.
CO2: Analyze the various optimization techniques for writing parallel high performance applications.
CO3: Discuss the basics of Parallel computers.
CO4: Learn shared memory parallel programming using OpenMP.
CO5: Understand the distributed memory parallel programming with MPI.

TEXT BOOK:

1. Georg Hager, Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall / CRC Computational Science series, 2011.

REFERENCES:

1. Robert Robey and Yuliana Zamora, Parallel and High Performance Computing, Manning Publications, 2021.
2. Thomas Sterling, Matthew Anderson, Maciej Brodowicz, High Performance Computing: Modern Systems and Practices, Morgan Kaufmann Publishers, 2018.

24AM922	QUANTUM COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Analyse the behaviour of basic quantum algorithms.
- Discuss simple quantum algorithms and information channels in the quantum circuit model.
- Apply the quantum algorithms in superdense coding and quantum teleportation.
- Analyse the algorithms with super-polynomial speed-up.
- Illustrate a simple quantum error-correcting code.

UNIT I	FOUNDATION	9

Overview of traditional computing – Church-Turing thesis – circuit model of computation – reversible computation – quantum physics – quantum physics and computation – Dirac notation and Hilbert Spaces – dual vectors – operators – the spectral theorem – functions of operators – tensor products – Schmidt decomposition theorem.

UNIT II	QUBITS AND QUANTUM MODEL OF COMPUTATION	9

State of a quantum system – time evolution of a closed system – composite systems – measurement – mixed states and general quantum operations – quantum circuit model – quantum gates – universal sets of quantum gates – unitary transformations – quantum circuits.

UNIT III	QUANTUM ALGORITHMS-I	9

Superdense coding – quantum teleportation – applications of teleportation – probabilistic versus quantum algorithms – phase kick-back – the Deutsch algorithm – the Deutsch- Jozsa algorithm – Simon's algorithm – Quantum phase estimation and quantum Fourier Transform – eigenvalue estimation.

UNIT IV	QUANTUM ALGORITHMS – II	9

Order-finding problem – eigenvalue estimation approach to order finding – Shor's algorithm for order finding – finding discrete logarithms – hidden subgroups – Grover's quantum search algorithm – amplitude amplification – quantum amplitude estimation – quantum counting – searching without knowing the success probability.

UNIT V	QUANTUM COMPUTATIONAL COMPLEXITY AND ERROR CORRECTION	9

Computational complexity – black-box model – lower bounds for searching – general black-box lower bounds – polynomial method – block sensitivity – adversary methods – classical error correction – classical three-bit code – fault tolerance – quantum error correction – three- and nine-qubit quantum codes – fault-tolerant quantum computation.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Analyse the behaviour of basic quantum algorithms.
CO2: Discuss simple quantum algorithms and information channels in the quantum circuit model.
CO3: Apply the quantum algorithms in superdense coding and quantum Teleportation.
CO4: Analyse the algorithms with super polynomial speed-up.
CO5: Illustrate a simple quantum error-correcting code.
CO6: Elaborate various quantum alogirthms.

TEXT BOOKS:

1. P. Kaye, R. Laflamme, and M. Mosca, "An introduction to Quantum Computing", Oxford University Press, 2007.
2. E. Rieffel and W. Polak, "Quantum Computing A Gentle Introduction", The MIT Press Cambridge, 2011.

REFERENCES:

1. Jack D. Hidary "Quantum Computing: An Applied Approach", Springer, 2019.
2. V. Sahni, "Quantum Computing", Tata McGraw-Hill Publishing Company, 2007.
3. Michael A. Nielsen and Issac L. Chuang, "Quantum Computation and Quantum Information", Tenth Edition, Cambridge University Press, 2010.

24AM923	FUNDAMENTALS OF ACCELERATED COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

1. Understand the fundamentals of heterogeneous parallel computing and GPU architecture.
2. Learn the CUDA programming model for writing GPU-accelerated applications.
3. Apply parallel computation patterns such as reduction, scan, and stencil to solve computational problems.
4. Analyze and optimize memory access patterns and kernel performance on GPU hardware.
5. Develop accelerated computing solutions using CUDA libraries and advanced parallel programming techniques.

UNIT I	INTRODUCTION TO HETEROGENEOUS PARALLEL COMPUTING	9

Introduction to heterogeneous parallel computing – Why more speed or parallelism – Speeding up real applications – Challenges in parallel programming – Related parallel programming interfaces – Heterogeneous data parallel computing – Data parallelism – CUDA C program structure – A vector addition kernel – Device global memory and data transfer – Kernel functions and threading – Calling kernel functions – Compilation.

UNIT II	GPU ARCHITECTURE AND MEMORY HIERARCHY	9

Multidimensional grids and data – Multidimensional grid organization – Mapping threads to multidimensional data – Image blur: a more complex kernel – Matrix multiplication – Compute architecture and scheduling – Architecture of a modern GPU – Block scheduling – Synchronization and transparent scalability – Warps and SIMD hardware – Control divergence – Warp scheduling and latency tolerance – Resource partitioning and occupancy – Querying device properties.

UNIT III	MEMORY OPTIMIZATION AND PERFORMANCE	9

Memory architecture and data locality – Importance of memory access efficiency – CUDA memory types – Tiling for reduced memory traffic – A tiled matrix multiplication kernel – Boundary checks – Impact of memory usage on occupancy – Performance considerations – Memory coalescing – Hiding memory latency – Thread coarsening – A checklist of optimizations – Knowing your computation's bottleneck.

UNIT IV	PARALLEL COMPUTATION PATTERNS	9

Convolution – Parallel convolution: a basic algorithm – Constant memory and caching – Tiled convolution with halo cells – Stencil – Shared memory tiling for stencil sweep – Parallel histogram – Atomic operations – Privatization – Reduction – Reduction trees – Minimizing control divergence – Minimizing memory divergence – Prefix sum (scan) – Kogge-Stone algorithm – Brent-Kung algorithm – Work efficiency considerations.

UNIT V	ADVANCED TOPICS AND CUDA LIBRARIES	9

Parallel sorting – Radix sort – Parallel merge sort – Sparse matrix computation – COO, CSR, and ELL formats – SpMV kernels – Graph traversal – Breadth-first search on GPU – Programming a heterogeneous computing cluster – CUDA streams and concurrency – Overlapping computation and communication – CUDA dynamic parallelism – Using CUDA libraries: cuBLAS, cuFFT, cuSOLVER, and Thrust – Parallel programming and computational thinking – Numerical considerations: floating-point precision and accuracy.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the architecture of modern GPUs and the CUDA programming model for heterogeneous computing. (K2)

CO2: Write CUDA C/C++ programs with appropriate thread hierarchy, memory management, and error handling for data-parallel problems. (K3)

CO3: Apply tiling, memory coalescing, and occupancy optimization techniques to improve GPU kernel performance. (K3)

CO4: Analyze the performance of GPU-accelerated applications using profiling tools and identify

computational bottlenecks. (K4)

CO5: Design and implement efficient parallel algorithms using advanced computation patterns and CUDA libraries for real-world applications. (K5)

CO6: Evaluate and select appropriate parallel computation patterns and memory optimization strategies for a given computational problem to achieve optimal GPU performance. **(K6)**

TEXT BOOKS:

1. Wen-mei W. Hwu, David B. Kirk, and Izzat El Hajj, "Programming Massively Parallel Processors: A Hands-on Approach," 4th Edition, Morgan Kaufmann (Elsevier), 2022.

2. John Cheng, Max Grossman, and Ty McKercher, "Professional CUDA C Programming," Wiley (Wrox), 2014.

REFERENCES:

1. Jason Sanders and Edward Kandrot, "CUDA by Example: An Introduction to General-Purpose GPU Programming," Addison-Wesley, 2010.

2. Shane Cook, "CUDA Programming: A Developer's Guide to Parallel Computing with GPUs," Morgan Kaufmann, 2013.

3. NVIDIA Deep Learning Institute (DLI), "Fundamentals of Accelerated Computing with CUDA C/C++," Online Course, https://learn.nvidia.com/

4. NVIDIA Deep Learning Institute (DLI), "Fundamentals of Accelerated Computing with CUDA Python," Online Course, https://learn.nvidia.com/

5. NVIDIA CUDA C++ Programming Guide, https://docs.nvidia.com/cuda/cuda-c-programming-guide/

24AM960	NATURE INSPIRED COMPUTING TECHNIQUES	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamentals of nature inspired techniques which influence computing
- To learn the computing inspired by nature
- To study the Swarm Intelligence
- To know about Immuno computing techniques
- To familiarize with DNA Computing

UNIT I	INTRODUCTION				9

From Nature to Natural Computing – Philosophy - Three Branches: Overview - Conceptualization - Individuals, Entities and agents - Parallelism and Distributivity -Interactivity, Adaptation- Feedback- Self-Organization-Complexity, Emergence and Reductionism - Bottom-up Vs Top-Down- Determination, Chaos and Fractals.

UNIT II	COMPUTING INSPIRED BY NATURE				9

Evolutionary Computing - Hill Climbing and Simulated Annealing - Darwin's Dangerous Idea - Genetics Principles - Standard Evolutionary Algorithm -Genetic Algorithms – Crossover – Mutation - Evolutionary Programming - Genetic Programming.

UNIT III	SWARM INTELLIGENCE				9

Introduction - Ant Colonies - Ant Foraging Behavior - Ant Colony Optimization, S-ACO Algorithm - Scope of ACO algorithms - Ant Clustering Algorithm (ACA) - Swarm Robotics -Foraging for food - Social Adaptation of Knowledge - Particle Swarm and Particle Swarm Optimization (PSO).

UNIT IV	IMMUNOCOMPUTING				9

Introduction- Immune System - Physiology and main components - Pattern Recognition and Binding - The Immune Network Theory- Danger Theory - Evaluation Interactions - Immune algorithms - Bone Marrow Models - Forrest's Algorithm - Artificial Immune Networks.

UNIT V	COMPUTING WITH NEW NATURAL MATERIALS				9

DNA Computing: Introduction - The DNA Molecule – Manipulating DNA - Adleman's experiment - Test tube programming language - Universal DNA Computers - PAM Model - Splicing Systems - Lipton's Solution to SAT Problem - Scope of DNA Computing - From Classical to DNA Computing.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics Natural systems.
CO2: Analyze the concepts of Natural systems and its applications.
CO3: Learn Ant Colony Optimization and Swarm Robotics.
CO4: Articulate immune algorithms and Artificial immune networks.
CO5: Learn DNA Molecule and Scope of DNA computing.

TEXT BOOKS:

1. Leandro Nunes de Castro, "Fundamentals of Natural Computing, Basic Concepts, Algorithms and Applications", Chapman and Hall/ CRC, Taylor and Francis Group, 2007.

24AM924	SCALABLE MACHINE LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Discuss the basics of how distributed computing is applied in scaling up machine learning process.
- Use scalable machine learning frameworks for parallel learning.
- Apply parallel Machine Learning Algorithms that can scale up.
- Distinguish traditional ML algorithms and Scalable ML algorithms.
- Discuss alternative learning for scalability.
- Solve Large-scale real-world problems using GPUs and Multi-core systems.

UNIT I	INTRODUCTION	9

Scaling Up – Reasons – Key Concepts – Platforms – Distributed Machine Learning – Stages of ML Workflow – Tools and Technologies in ML Pipeline – Distributed Computing Models – Distributed Systems Architecture – Ensemble Models – Challenges.

UNIT II	FRAMEWORKS FOR SCALLING UP	9

Apache Spark Architecture – PySpark – MapReduce for Massively Parallel Learning – Uniformly Fine-Grained Data-Parallel Computing – GP-GPU.

UNIT III	LEARNING ALGORITHMS	9

PSVM: Parallel Support Vector Machines with Incomplete Cholesky Factorization - PSVM Algorithm - Massive SVM Parallelization Using Hardware Accelerators - SMO Algorithm - Large-Scale Learning to Rank Using Boosted Decision Trees – LambdaMART - Large-Scale Spectral Clustering with MapReduce and MPI.

UNIT IV	ALTERNATIVE LEARNING	9

Parallel Online Learning - Limits Due to Bandwidth and Latency - Parallelization Strategies - Delayed Update Analysis - Parallel Learning Algorithms - Global Update Rules - Distributed Transfer Learning via Cooperative Matrix Factorization - Distributed Coalitional Learning - Extension of DisCo to Classification Tasks - Parallel Large-Scale Feature Selection.

UNIT V	APPLICATIONS	9

Large-Scale Learning for Vision with GPUs - Standard Pipeline – GPUs – Approach - Feature Learning with Deep Belief Networks - Mining Tree-Structured Data on Multicore Systems - Multicore Challenge - Memory Optimizations - Adaptive Parallelization - Empirical Evaluation.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Discuss the basics of how distributed computing is applied in scaling up machine learning process.
CO2: Use scalable machine learning frameworks for parallel learning.
CO3: Apply parallel Machine Learning Algorithms that can scale up.
CO4: Distinguish traditional ML algorithms and Scalable ML algorithms.
CO5: Discuss alternative learning for scalability.
CO6: Solve Large-scale real-world problems using GPUs and Multi-core systems.

TEXT BOOKS:

1. Ron Bekkerman, Mikhail Bilenko and John Langford, Scaling Up Machine Learning: Parallel and Distributed Approaches, Cambridge University Press, 2012.
2. Adi Polak, Scaling Machine Learning with Spark, O'Reilly Media, 2023.

REFERENCES:

1. J. Joshua Thomas, S. Harini, V. Pattabiraman, Scalable and Distributed Machine Learning and Deep Learning Patterns (Advances in Computational Intelligence and Robotics), IGI Global, 2023.
2. Bastiaan Sjardin, Luca Massaron, Alberto Boschetti, Large Scale Machine Learning with Python, Packt Publications, 2016.

24AM925	OPTIMIZATION METHODS IN MACHINE LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the basics of different Submodular functions and Associated Ployhedra.
- To discuss Submodularity and its Applications.
- To analyze the various methods of Non-Smooth Convex Optimizations.
- To analyze the various Separable Optimization Problems.
- To discuss the various Submodular minimization methods and optimizations.

UNIT I	INTRODUCTION	9

Introduction – Definition – Submodularity – Associated Polyhedra – Polymatroids – Lovasz Extension – Definition – Greedy Algorithm – Links between submodularity and convexity. Properties of Associated Polyhedra: Support functions – Facial Structure – Positive and Symmetric submodular Polyhedra.

UNIT II	SUBMODULARITY	9

Convex and Concave closures of set functions – Structured Sparsity – Convex Relaxation of Combinatorial Penalty – l_q relaxations of submodular penalties – Shaping level sets – Examples and Applications of Submodularity – Cardinality based functions – Cut functions – Set Covers – Flows – Entropies – Spectral functions of submatrices – Best Subset Selection – Matroids.

UNIT III	NON-SMOOTH CONVEX OPTIMIZATION	9

Projected Subgradient descent – Ellipsoid Method – Kelly's Method – Analytic Centre Cutting planes – Mirror descent/conditional gradient – Bundle and Simplicial Methods – Proximal Methods – Simplex algorithm for Linear Programming – Active Set Method for Quadratic Programming – Active Set Algorithms for Least-squares Problems.

UNIT IV	SEPARABLE OPTIMIZATION PROBLEMS	9

Analysis: Optimality conditions for base polyhedral – Equivalence with submodular function Minimization – Quadratic Optimization Problems – Separable problems on other polyhedra.
Algorithms: Divide-and Conquer algorithm for proximal problems – Iterative algorithms – Exact minimization-Approximate minimization.

UNIT V	SUBMODULAR MINIMIZATION AND OPTIMIZATION	9

Minimizers of Submodular Functions – Combinatorial Algorithms – Minimizing Symmetric posimodular functions – Ellipsoid method – Simplex method for Submodular function minimization – Analytic centre cutting planes -Minimum norm point algorithm – Approximate minimization through convex optimization – Special Structure. Maximization with cardinality constraints – Submodular function minimization.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Understand the basics of different Submodular functions and Associated Ployhedra.
CO2: Discuss Submodularity and its Applications.
CO3: Analyze the various methods of Non-Smooth Convex Optimizations.
CO4: Analyze the various Separable Optimization Problems.
CO5: Discuss the various Submodular minimization methods and optimizations.
CO6: Apply various optimization methods to solve real-world problem in machine learning.

TEXT BOOKS:

1. Francis Bach, "Learning with Submodular Functions: A Convex Optimization Perspective", Foundations and Trends in Machine Learning, Now Publishers Inc., 2013.

REFERENCES:

1. A. Beck, "First-Order Methods in Optimization", MOS-SIAM Series on Optimization, 2017.
2. S. Bubeck, "Convex Optimization: Algorithms and Complexity, Foundations and Trends in Optimization", 2015.
3. Stephen Boyd, Lieven Vandenberghe, Convex Optimization, Cambridge University Press, Seventh Edition, 2009.
4. Suvrit Sra, Sebastian Nowozin, and Stephen J. Wright, Optimization for Machine Learning

24AM926	APPLIED ACCELERATED ARTIFICIAL INTELLIGENCE	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

1. Understand the fundamentals of deep learning architectures and GPU acceleration for AI workloads.
2. Build and train deep learning models using GPU-accelerated frameworks such as TensorFlow and PyTorch.
3. Apply transfer learning, fine-tuning, and GPU-accelerated vector search techniques for advanced deep learning tasks.
4. Implement advanced GPU programming concepts including CUDA programming model, memory hierarchy, and optimization techniques.
5. Design and deploy distributed and multi-GPU training strategies for scalable AI solutions.

UNIT I	INTRODUCTION TO DEEP LEARNING AND GPU ACCELERATION	9

Introduction – Historical Background – Applications of Deep Learning – Deep Learning Architectures – Deep Neural Network – Linear Regression as a Single-Layer Neural Network – Convolutional Neural Networks (CNNs) – Evolving from Dense Layers to Convolutional Networks – Translation Invariance – Locality – Convolutions – Recurrent Neural Networks (RNNs) – Mathematical Representation of RNNs – Unrolling an RNN – Challenges in RNNs: Vanishing and Exploding Gradients – Long Short-Term Memory (LSTM) – Applications of RNNs – GPU Acceleration in Deep Learning – Introduction to GPU Acceleration – Types of GPU Acceleration – Training a Deep Learning Model – Tensors – Transforms – DataLoaders – Creating the Model – Compiling the Model – Calculating Accuracy.

UNIT II	GPU ARCHITECTURES, OPTIMIZATIONS, AND DEPLOYMENT	9

Deep Learning Frameworks – What Is a Deep Learning Framework? – Why Use a Deep Learning Framework? – Popular Frameworks: TensorFlow, PyTorch, JAX – Comparison of Popular Frameworks – Setting GPU Support – NVIDIA GPU Driver Requirements – Installing CUDA and cuDNN – Verifying GPU Availability – Managing GPU Memory Usage and Visibility – GPU Acceleration in TensorFlow – GPU-Compatible Tensors and Operations – Automatic vs. Manual Device Placement – Enabling Mixed-Precision Training – Using MirroredStrategy for Multi-GPU Training – Multi-GPU Training Strategies: Data Parallelism, Model Parallelism, Hybrid Parallelism – GPU Memory Optimization Techniques – Gradient Checkpointing (Recomputation) – Mixed-Precision Training – Layer-Wise Memory Reuse – Memory Growth and Garbage Collection – TensorRT Integration for Inference Optimization.

UNIT III	ADVANCED DEEP LEARNING ARCHITECTURES FOR GPU WORKLOADS	9

Transfer Learning with GPU Acceleration – Motivation and Need – Mathematical Model for Transfer Learning – Relevance with GPUs – GPU Benefits in Fine-Tuning Pretrained Models – Fine-Tuning vs. Training – Fine-Tuning Techniques: Freezing Bottom Layer, Parameter-Efficient Fine-Tuning (PEFT), Prompt Tuning – Role of GPU in Fine-Tuning – Forward Pass – Loss Function – Backpropagation and Gradient Update – Fine-Tuning in Computer Vision – GPU Acceleration in Deep Network Tuning – GPU-Accelerated Vector Search Indexing Methods: IVF-Flat, IVF-PQ, CAGRA – GPU Architecture for Deep Network Tuning – GPU Benefits in Fine-Tuning.

UNIT IV	ADVANCED GPU PROGRAMMING FOR DEEP LEARNING	9

CUDA Programming Model – What Is CUDA? – Heterogeneous Computing: Host and Device – Interaction Between Host and Device – CUDA Extensions to C/C++ – Function Launch and Kernel Execution – Calling Functions on the GPU – Memory Management – Memory Declaration Qualifiers – Thread and Block Identification Keywords – Function Declarations in CUDA – CUDA Function Qualifiers Overview – Variable Type Qualifiers in CUDA – SIMT Architecture – Mapping Blocks and Threads – Thread Life Cycle in a GPU – Device Memory Management – Memory Hierarchy – GPU Memory Management: Types of GPU Memory (Global, Shared, Constant, Texture) – Memory Growth, Garbage Collection, and Layer-wise Memory Reuse – Efficient Memory Access and Coalescing Strategies.

UNIT V	DISTRIBUTED TRAINING STRATEGIES AND EMERGING TRENDS	9

Distributed and Multi-GPU Training Strategies – Data Parallelism Strategies – Model Parallelism Strategies – Pipeline Parallelism – Multi-GPU Communication: NCCL, Gloo – TensorFlow Distributed Training: MirroredStrategy, MultiWorkerMirroredStrategy – Cluster Setup and Role Configuration – Monitoring with TensorBoard – Horovod and DeepSpeed – Horovod Overview and Integration – Zero Redundancy Optimizers in DeepSpeed – Cloud and HPC Considerations – Distributed Training on AWS/GCP/Azure – Running on HPC Infrastructure: Using SLURM and MPI

– Monitoring, Profiling, and Debugging Tools: NVIDIA Nsight and NVProf, TensorBoard and PyTorch Profiler – Recent Applications and Emerging Trends – Federated and Decentralized Learning – Energy-Efficient Multi-GPU Training – AI Model Training on Exascale Systems.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Describe deep learning architectures including CNNs, RNNs, and LSTMs, and explain the role of GPU acceleration in training deep learning models. (K2)

CO2: Build and train deep learning models using GPU-accelerated frameworks with appropriate device placement, mixed-precision training, and memory optimization techniques. (K3)

CO3: Apply transfer learning, fine-tuning techniques, and GPU-accelerated vector search methods for domain-specific deep learning tasks. (K3)

CO4: Analyze CUDA programming constructs including memory hierarchy, SIMT architecture, and coalescing strategies for optimizing GPU-based deep learning workloads. (K4)

CO5: Design and evaluate distributed multi-GPU training strategies using data parallelism, model parallelism, Horovod, and DeepSpeed for scalable AI applications. (K5)

TEXT BOOKS:

1. Ramchandra S. Mangrulkar and Pallavi Vijay Chavan, "GPU-Accelerated Deep Learning: Essential GPU Ideas, Deep Learning Frameworks, and Optimization Approaches," Apress (Springer), 2025.

2. Eli Stevens, Luca Antiga, and Thomas Viehmann, "Deep Learning with PyTorch," Manning Publications, 2020.

REFERENCES:

1. Jeremy Howard and Sylvain Gugger, "Deep Learning for Coders with fastai and PyTorch: AI Applications Without a PhD," O'Reilly Media, 2020.

2. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning," MIT Press, 2016. (Available online: https://www.deeplearningbook.org/)

3. NPTEL Course - "Applied Accelerated Artificial Intelligence," NPTEL Online Course, https://onlinecourses.nptel.ac.in/

4. NVIDIA Deep Learning Institute, "Fundamentals of Deep Learning" and "Fundamentals of Accelerated Data Science with RAPIDS," Online Courses, https://learn.nvidia.com/

5. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow," 3rd Edition, O'Reilly Media, 2022.

COMPUTATIONAL INTELLIGENCE

24AM951	COMPUTER NETWORKS AND SECURITY	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the fundamental concepts of computer networks.
- Explain the various protocols and techniques used in the physical layer and data link layer.
- Illustrate different protocols used in the network and transport layers.
- Analyze the various application layer protocols.
- Discuss the key concepts of network security.

UNIT I	PHYSICAL AND DATA LINK LAYER	9

Data Communications – Network Types – Protocol Layering – Network Models (OSI, TCP/IP) – Physical Layer – Multiplexing – Transmission Media – Data-Link Layer – Nodes and Links – Types – Sublayers – Data-link Control – Media Access Protocols – Link Layer Addressing – LANs – Ethernet- WiFi, IEEE 802.11, Bluetooth – Connecting Devices.

UNIT II	NETWORK LAYER	9

Services – Packet Switching – Performance – IPV4 Addressing – IPV4 Protocols – IPV6 Addressing – IPV6 Protocols – Routing Algorithms – Unicast Routing Protocols – Multicast Routing – IGMP.

UNIT III	TRANSPORT LAYER	9

Transport Layer Services - Transport Layer Protocols - User Datagram Protocol –Transmission Control Protocol – SCTP.

UNIT IV	APPLICATION LAYER	9

Providing Services – Application Layer Paradigms – Client/Server Paradigm – WWW and HTTP – FTP – Email – Telnet - SSH – DNS – SNMP.

UNIT V	CRYPTOGRAPHY AND NETWORK SECURITY	9

Attacks – Services and Techniques – Confidentiality – Message Integrity, Authentication – Digital Signature – Entity Authentication – Key Management – Network Layer Security – Transport Layer Security – Application Layer Security – Firewalls.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the fundamental concepts of computer networking and network architecture.
CO2: Analyze the performance of various network protocols used in data transmission.
CO3: Analyze and implement the various routing protocols.
CO4: Develop skills to diagnose common network issues using tools.
CO5: Analyze the various application layer protocols.
CO6: Elaborate the key concepts of cryptography and network security.

TEXT BOOK:

1. Behrouz A. Forouzan, Data Communications and Networking, McGraw Hill Education, 6th Edition, 2022.

REFERENCES:

1. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson education 8th Edition, 2023.
2. James F. Kurose, Amherst Keith Ross, Computer Networking - A Top Down Approach, 8th Edition, 2021.
3. Andrew S. Tanenbaum, Computer Networks, Sixth Edition, Pearson, 2021.
4. P.C. Gupta, Data Communications and Computer Networks, Prentice-Hall of India, 2006.
5. L. L. Peterson and B. S. Davie, Computer Networks: A Systems Approach, Morgan Kaufmann, 3rd ed., 2003.

6. C K Shyamala, N Harini and Dr. T R Padmanabhan: Cryptography and Network Security, Wiley India Pvt. Ltd., 1st Edition, 2011.
7. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, Tata Mc Graw Hill, 2015.
8. Nina Godbole, Sunit Belapure, “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, First Edition, Wiley India, 2011.

24CS929	SOFT COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To learn the basic concepts of Soft Computing.
- To understand artificial neural networks.
- To apply fuzzy systems to solve problems.
- To solve problems using Genetic Algorithms.
- To discuss the various Hybrid algorithms and various Swarm Intelligence algorithms.

UNIT I	INTRODUCTION	9

Neural Networks - Application Scope of Neural Networks - Fuzzy Logic - Genetic Algorithm - Hybrid Systems - Soft Computing - Artificial Neural Network - Evolution of Neural Networks - Basic Models of ANN – Weights – Bias – Threshold – Learning Rate – Momentum Factor – Vigilance Parameter- McCulloch–Pitts Neuron - Linear Separability - Hebb Network.

UNIT II	ARTIFICIAL NEURAL NETWORKS	9

Perceptron Networks - Adaptive Linear Neuron - Multiple Adaptive Linear Neurons - Back-Propagation Network - Radial Basis Function Network - Pattern Association – Auto associative and Hetero associative Memory Networks - Bidirectional Associative Memory (BAM) - Hopfield Networks - Fixed Weight Competitive Nets - Kohonen Self-Organizing Feature Maps.

UNIT III	FUZZY SYSTEMS	9

Fuzzy Logic - Classical Sets (Crisp Sets) - Fuzzy Sets – Fuzzy Relation - Features of the Membership Functions - Fuzzification - Methods of Membership Value Assignments - Defuzzification - Lambda-Cuts for Fuzzy Sets (Alpha-Cuts) - Lambda-Cuts for Fuzzy Relations - Defuzzification Methods – Fuzzy Reasoning – Fuzzy Inference Systems

UNIT IV	GENETIC ALGORITHMS	9

Biological Background - Traditional Optimization and Search Techniques- Genetic Algorithm and Search Space- - Simple GA - General Genetic Algorithm - Operators - Stopping Condition - Constraints - Problem Solving - The Schema Theorem- Classification - Holland Classifier Systems- Genetic Programming - Advantages and Limitations- Applications.

UNIT V	HYBRID SOFT COMPUTING AND SWARM INTELLIGENCE ALGORITHMS	9

Neuro-Fuzzy Hybrid Systems - Genetic Neuro-Hybrid Systems - Genetic Fuzzy Hybrid and Fuzzy Genetic Hybrid Systems - Simplified Fuzzy ARTMAP – Swarm Intelligence Algorithms - Ant Colony Optimization – Artificial Bee Colony – Particle Swarm Optimization – Firefly Algorithm.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the basic concepts of Soft Computing.
CO2: Discuss Artificial neural networks and its applications.
CO3: Apply Fuzzy logic to solve different applications.
CO4: Solving problems using Genetic algorithms.
CO5: Discuss various algorithms in Soft computing with its applications and limitations.
CO6: Use various algorithms in Soft computing to solve real-world problems.

TEXT BOOKS:

1. S. N. Sivanandam , S. N. Deepa, "Principles of Soft Computing", Wiley India Pvt. Ltd., 2nd Edition, 2019.
2. Adam Slovik, “Swarm Intelligence Algorithms: Modification and Applications”, Taylor & Francis, First Edition, 2020.

REFERENCES:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.
2. Kwang H. Lee, First course on Fuzzy Theory and Applications‖, Springer, 2005.
3. N.P. Padhy, S. P. Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.

4. S. Rajasekaran, G. A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications ", PHI Learning Pvt. Ltd., 2017.
5. NPTEL Courses:
    a. Introduction To Soft Computing - https://onlinecourses.nptel.ac.in/noc23_cs40/preview

24AM930	EVOLUTIONARY AND SWARM INTELLIGENCE	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Introduce the fundamental concepts of **Evolutionary Computation** and nature-inspired problem solving.
- Develop proficiency in applying the Genetic Algorithm and its operators to solve complex optimization problems.
- Expose students to advanced evolutionary techniques such as **Evolutionary Strategies** and **Genetic Programming**.
- Understand the principles of **Swarm Intelligence** and its decentralized nature.
- Enable the application of swarm-based algorithms like **Particle Swarm Optimization** and **Ant Colony Optimization** in real-world problems.

UNIT I	FUNDAMENTALS OF EVOLUTIONARY COMPUTATION	9

Introduction to Evolutionary Computation - Biological inspiration: selection, mutation, reproduction - Evolutionary process cycle - Fitness function and search space - Representation of solutions (encoding techniques) - Basic optimization concepts – Population-based search and termination criteria

UNIT II	GENETIC ALGORITHMS	9

Concept of Genetic Algorithm - Structure of GA and working principle - Chromosome encoding techniques - Genetic operators: Selection (Roulette wheel, Tournament), Crossover (single-point, multi-point, uniform), Variation Techniques including mutation - Elitism and replacement strategies - Convergence and performance considerations – Real - world applications of GA.

UNIT III	ADVANCED EVOLUTIONARY TECHNIQUES	9

Evolutionary Strategies – concepts and working - $(\mu + \lambda)$ and (μ, λ) selection strategies - Adaptation and parameter control techniques - Genetic Programming - Tree representation, function & terminal sets - Fitness evaluation and operators in GP - Comparison of evolutionary methods.

UNIT IV	SWARM INTELLIGENCE AND PSO	9

Introduction to Swarm Intelligence - Characteristics: self-organization, decentralization - Natural swarm examples (bird flocking, fish schooling) - Particle Swarm Optimization - Particle representation and initialization - Velocity and position update equations - Global best (gbest) and local best (lbest) - Parameters (inertia weight, cognitive & social factors) - Applications of PSO

UNIT V	ANT COLONY OPTIMIZATION AND APPLICATIONS	9

Ant Colony Optimization - Biological inspiration (pheromone trails) - Path construction and probabilistic transition - Pheromone update and evaporation - Applications (Traveling Salesman Problem) - Overview of other swarm methods: Artificial Bee Colony, Firefly Algorithm - Applications (TSP, routing, scheduling, ML optimization)

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the principles of evolutionary and swarm-based optimization techniques.

CO2: Apply Genetic Algorithm to solve complex optimization problems.

CO3: Analyze and compare different evolutionary methods such as Evolutionary Strategies and Genetic Programming.

CO4: Implement swarm intelligence techniques like Particle Swarm Optimization and Ant Colony Optimization.

CO5: Design and develop solutions for real-world applications using evolutionary and swarm intelligence approaches.

TEXT BOOKS:

1. An Introduction to Genetic Algorithms – Melanie Mitchell, Publisher: MIT Press , Year: 1998, Edition: 1st Edition
2. Evolutionary Computation A Unified Approach – Thomas Bäck, David B. Fogel, Zbigniew Michalewicz, Publisher: MIT Press, Year: 2000, Edition: 1st Edition
3. Swarm Intelligence – James Kennedy, Russell Eberhart, Publisher: Morgan Kaufmann, Year:

2001, Edition: 1st Edition

REFERENCES:

1. Genetic Algorithms in Search Optimization and Machine Learning – David E. Goldberg, Addison-Wesley, 1989
2. Ant Colony Optimization – Marco Dorigo and Thomas Stützle, MIT Press, 2004
3. Particle Swarm Optimization – Maurice Clerc, ISTE Ltd / John Wiley & Sons, 2006.
4. Swarm Intelligence – James Kennedy and Russell Eberhart, Morgan Kaufmann, 2001.
5. Artificial Intelligence A Modern Approach – Stuart Russell and Peter Norvig, Pearson, 2021 (4th Edition).

24AM928	APPLIED AI and ML	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand and apply statistical methods to analyze and interpret data.
- Analyze and cluster genomic data using appropriate algorithms.
- Implement linear regression models to predict outcomes.
- Evaluate and improve model performance in binary classification tasks.
- Implement and train neural networks for various tasks.

UNIT I	FOUNDATION OF DATA SCIENCE	9

Python for Data Science- NumPy & Pandas - Data Cleaning and Preparation- Statistics for Data Science- Types of Data- Levels of Measurement-Descriptive Statistics-Probability theory -Inferential Statistics-Advanced Visualization Techniques.

Case Study: Cardio Good Fitness Data Analysis

Projects:1. Food Hub Analysis
2. FIFO World Cup Analysis
3. Mobile Internet Usage Analysis

UNIT II	MAKING SENSE OF UNSTRUCTURED DATA	9

Introduction to Supervised & Unsupervised Learning- Handling Imbalanced Datasets-K-Means Clustering algorithm, Dimensionality Reduction techniques (PCA, t-SNE)-Visualizing High Dimensional Data-Comparsion of t-SNE with PCA-Combining PCA with t-SNE.

Case Study: Genomic Data Clustering

Project: Fantasy Sports Clustering Analysis

UNIT III	REGRESSION AND PREDICTION	9

Introduction to Linear Regression-OLS Method-Cost function and Optimization-Gradient Descent Algorithm-Multiple Linear Regression-Elastic Net, Model Evaluation Techniques in solving Real World Regression Problems.

Case Studies: 1. Hospital LOS Prediction
2.Big Mart Sales Prediction

Project: Super Kart Sales Prediction

UNIT IV	CLASSIFICATION AND HYPOTHESIS TESTING	9

Concepts of Classification algorithms- Model Performance- Application of Binary Classification-Multi class classification-Multi label classification-Challenges in solving real world classification problems.

Case Studies: 1.HR Employee Attrition Prediction
2. KC Roasters Coffee Quality Prediction

Projects: 1. Travel Package Purchase Prediction
2. Potential Customers Prediction

UNIT V	DEEP LEARNING	9

Implementation of Neural Networks-Data Quality & Quantity-Data Augmentation- Hyper parameter tuning-Computational Challenges -Transformer Networks-Transfer learning -solving real world Neural Network based Problems.

Case Study: 1. Audio MNLST Digit Recognition,
2.Street View Housing Number Digit Recognition

Project: Food Image Classification

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Apply statistical techniques to interpret data and make data-driven decisions.

CO2: Utilize dimensionality reduction techniques such as PCA and t-SNE to simplify complex datasets.

CO3: Apply regression techniques to real-world problems.

CO4: Perform hypothesis testing to validate assumptions and make inferences from data.
CO5: Apply deep learning techniques to solve practical problems.
CO6: Implement the concepts of AI and ML to solve various applications.

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, Machine Learning, Pearson, 2019.
2. Ethem Alpaydin, Introduction to Machine Learning, Adaptive Computation and Machine Learning Series, Third Edition, MIT Press, 2014.
3. Deep Learning: A Practitioner's Approach, Josh Patterson, Adam Gibson, O'Reilly Media, 2017.
4. Deep Learning, Ian Goodfellow, Yoshua Bengio Aaron Courville, MIT Press, 2017.
5. Neural Networks and Deep Learning, Michael Nielsen, Determination Press, 2015.

REFERENCES:

1. Anuradha Srinivasaraghavan, Vincy Joseph, Machine Learning, First Edition, Wiley, 2019.
2. Peter Harrington, “Machine Learning in Action”, Manning Publications, 2012.
3. Stephen Marsland, “Machine Learning – An Algorithmic Perspective”, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
4. Tom M Mitchell, Machine Learning, First Edition, McGraw Hill Education, 2013.
5. Christoph Molnar, “Interpretable Machine Learning - A Guide for Making Black Box Models Explainable”, Creative Commons License, 2020.
6. Deep Learning with TensorFlow: Explore neural networks with Python, Giancarlo Zaccone, Md. Rezaul Karim, Ahmed Menshawy, Packt Publisher, 2017.
7. Deep Learning with Keras, Antonio Gulli, SujitPal , Packt Publishers, 2017.
8. Deep Learning with Python", Francois Chollet, Manning Publications, 2017
9. https://olympus.mygreatlearning.com/courses

24AM931	AFFECTIVE COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- Understand the history, evolution, and key technologies of human–computer interaction.
- Learn the psychological and neuroscience foundations of emotion and how emotions are modeled computationally.
- Understand how emotions are recognized from visual cues (facial expressions, body language) and voice (speech signals).
- Study brain–computer interface technology and emotion recognition from physiological electrical signals.
- Explore the design and emotional interaction technology of social robots.
- Evaluate the current challenges, ethical issues, and future trends in affective computing.

UNIT I	HUMAN–COMPUTER INTERACTION AND SCIENTIFIC BASIS OF AFFECTIVE COMPUTING (Ch 1–2)	9

History and Development of Human–Computer Interaction: The Evolution of Human–Computer Interaction - The Evolution of Human–Computer Interaction. - Key Technologies and Equipment for Human–Computer Interaction - Overview of Modern Interaction Technologies - Concept and Goal of Human–Machine Integration.

Scientific Basis of Affective Computing: Definition and Model of Affective Computing - Psychological Basis of Emotion - Neuroscience Basis of Emotion - Mechanisms of Emotion Recognition and Expression.

UNIT II	VISUAL INTERACTION TECHNOLOGY AND VOICE INTERACTION TECHNOLOGY (Ch 3–4)	9

Visual Interaction Technology: Principles of Visual Perception - Facial Expression Recognition. - Body Language Recognition.

Voice Interaction Technology: Speech Recognition and Synthesis - Emotional Speech Analysis.

UNIT III	BRAIN–COMPUTER INTERFACE TECHNOLOGY AND PHYSIOLOGICAL ELECTRICAL SIGNALS (Ch 5–6)	9

Brain–Computer Interface Technology: Working Principle of Brain–Computer Interface - Application of Brain–Computer Interface in Affective Computing.

Emotional Computing of Physiological Electrical Signals: Types and Characteristics of Physiological Electrical Signals - Physiological Signal Processing Methods - Sentiment Analysis of Physiological Electrical Signals.

UNIT IV	SOCIAL ROBOTS: DESIGN PRINCIPLES AND EMOTIONAL INTERACTION (Ch 7–8)	9

Design Principles for Social Robots: Interaction Model of Social Robots - Personalization of Social Robots.

Emotional Interaction Technology of Social Robots: Implementation Mechanism of Emotional Interaction. - Application Case Analysis of Emotional Interaction – Education - Medical Field - Service Industry - Entertainment and Social Fields - Military and Security Field - Other Fields.

UNIT V	CURRENT CHALLENGES AND FUTURE DEVELOPMENT TRENDS (Ch 9–10)	9

Current Challenges: Technical Challenges - Ethical and Social Challenges.

Future Development Trends: Technology Development Trends - Social and Cultural Influences.

TOTAL: 45 PERIODS

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

CO1: Explain the evolution of human–computer interaction and the psychological and neuroscience foundations of emotion. *[K2 – Understand]*

CO2: Apply visual interaction techniques including facial expression recognition and body language recognition for emotion detection. *[K3 – Apply]*

CO3: Apply speech signal processing techniques to perform emotional speech analysis and voice-based emotion recognition. *[K3 – Apply]*

CO4: Analyze EEG and physiological electrical signals to classify emotional states using

appropriate signal processing and machine learning methods. *[K4 – Analyze]*

CO5: Design a social robot or emotion-aware interactive system by integrating multimodal emotion recognition and response technologies. *[K6 – Create]*

CO6: Evaluate the technical challenges, ethical implications, and future trends of affective computing in real-world applications. *[K5 – Evaluate]*

TEXT BOOKS:

1. Changzeng Fu, "Foundation of Affective Computing and Interaction", HACI Lab, Northeastern University, 2025. Creative Commons BY-NC 4.0 License. Freely available online: https://arxiv.org/abs/2506.15497

REFERENCES:

1. Rosalind W. Picard, "Affective Computing", MIT Press, 1997. ISBN: 9780262661157.
2. Rafael A. Calvo, Sidney D'Mello, Jonathan Gratch, and Arvid Kappas (Eds.), "The Oxford Handbook of Affective Computing", Oxford University Press, 2015. ISBN: 9780199942237.
3. Leimin Tian, Sharon Oviatt, and Michal Muszynski, "Applied Affective Computing", ACM Books, 2022. ISBN: 9781450395915.
4. Jenny Preece, Helen Sharp, and Yvonne Rogers, "Interaction Design: Beyond Human–Computer Interaction", 5th Edition, Wiley, 2019. ISBN: 9781119547259.
5. The Encyclopedia of Human–Computer Interaction, 2nd Edition, Interaction Design Foundation. Freely available at https://www.interaction-design.org/literature
6. IEEE Transactions on Affective Computing (Journal). Available at https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5165369

24AM932	RECOMMENDER SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand the foundations of the recommender system.
- To learn about collaborative filtering.
- To discuss content-based recommendation systems.
- To elaborate on the evaluation paradigms for a recommendation system.
- To make students design and implement a recommender system.

UNIT I	INTRODUCTION TO RECOMMENDER SYSTEMS	9

Introduction - Basic Models of Recommender Systems - Domain-Specific Challenges in Recommender Systems - Cold-Start Problem – Attack-Resistant Recommender Systems – Group – Multi-Criteria – Active-Learning – Privacy - Application Domains.

UNIT II	COLLABORATIVE FILTERING	9

Neighborhood-Based Collaborative Filtering - Key Properties - Predicting Ratings – Clustering - Dimensionality Reduction - A Regression Modeling - Graph Models – Model-based Collaborative Filtering - Decision and Regression Trees - Rule-Based Collaborative Filtering - Naive Bayes Collaborative Filtering – Latent Factor Models.

UNIT III	CONTENT-BASED RECOMMENDATION	9

Basic Components of Content-Based Systems - Preprocessing and Feature Extraction - Learning User Profiles and Filtering - Content-Based Versus Collaborative Recommendations - Using Content-Based Models for Collaborative Filtering.

UNIT IV	DESIGN EVALUATION	9

Evaluating Paradigms – General Goals of Evaluation Design-Design Issues in Offline Recommender Evaluation-Accuracy Metrics in Offline Evaluation-Limitations of Evaluation Measures.

UNIT V	TYPES OF RECOMMENDATION SYSTEMS	9

Content-based Recommender Systems – Basic Components – Constraint-based Recommender Systems – Context-sensitive Recommender Systems – Social and Trust-Centric Recommender Systems.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the foundations of the recommender system.
CO2: Use collaborative filtering to design recommendation systems.
CO3: Discuss content-based recommendation systems.
CO4: Elaborate on the evaluation paradigms for a recommendation system.
CO5: Use appropriate type of recommendation systems to solve real-world problems.
CO6: Design, implement and evaluate a recommendation algorithm.

TEXT BOOKS:

1. Charu C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.
2. Jannach D., Zanker M., FelFering A., Friedrich G., Recommender Systems: An Introduction, Cambridge University Press, First Edition, 2011.

REFERENCES:

1. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Mining of massive datasets, 3rd edition, Cambridge University Press, 2020.
2. Ricci, F., Rokach, L. and Shapira, B., Introduction to recommender systems handbook. In Recommender systems handbook, Springer, 2011.
3. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer, First Edition, 2013.

24AM933	NEUROMORPHIC COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES: **The Course will enable learners to:**

1. To introduce the fundamental concepts of neuromorphic computing and biological neural systems.
2. To understand neuron models, spiking neural networks, and their hardware design fundamentals.
3. To elaborate learning mechanisms and emerging memory technologies for neuromorphic systems.
4. To introduce communication network architectures and fault-tolerant design methods for neuromorphic systems.
5. To understand reconfigurable neuromorphic architectures and survey existing neuromorphic platforms.

UNIT I	INTRODUCTION TO NEUROMORPHIC COMPUTING SYSTEMS	9

Introduction to Neuromorphic Computing Systems – From Biological Brains to Artificial Brains – What Is Neuromorphic Computing? – Need for Neuromorphic Computing – The Von Neumann Bottleneck – Moore's Law and Its Limitations – Overview of the Brain Architecture – Key Computational Principles of the Brain. Neuromorphic System Design Fundamentals – Artificial Neural Network Concepts – Perceptron – Multilayer Perceptron – Activation Functions – Deep Neural Networks – Convolutional Neural Networks.

UNIT II	SPIKING NEURAL NETWORKS AND NEURON MODELS	9

Neuromorphic System Design Fundamentals (continued) – From Artificial Neural Networks to Spiking Neural Networks – Biological Neuron and Synapse – Spike Encoding – Rate Coding – Temporal Coding – Population Coding. Neuron Models – The Hodgkin–Huxley Model – Integrate-and-Fire Model – Leaky Integrate-and-Fire Model – Izhikevich Model – Spike Response Model – Digital Neuron Implementation – Analog Neuron Implementation.

UNIT III	LEARNING IN NEUROMORPHIC SYSTEMS AND EMERGING MEMORY DEVICES	9

Learning in Neuromorphic Systems – Synaptic Plasticity – Hebbian Learning – Spike-Timing-Dependent Plasticity (STDP) – Supervised Learning in Spiking Neural Networks – Unsupervised Learning in Spiking Neural Networks – Reinforcement Learning in Spiking Neural Networks – Hardware Implementation of Learning Rules. Emerging Memory Devices for Neuromorphic Systems – Memristor – Resistive Random Access Memory (ReRAM) – Phase Change Memory (PCM) – Spin-Transfer Torque Magnetic RAM (STT-MRAM) – Ferroelectric FET (FeFET) – Crossbar Array Architecture.

UNIT IV	COMMUNICATION NETWORKS AND FAULT-TOLERANT DESIGN	9

Communication Networks for Neuromorphic Systems – Address Event Representation (AER) Protocol – Network-on-Chip (NoC) Architecture – NoC Routing Algorithms – Interconnect Design Principles – On-Chip and Off-Chip Communication – Photonic Interconnects for Neuromorphic Systems. Fault-Tolerant Neuromorphic System Design – Reliability Threats in Neuromorphic Systems – Fault Models – Fault Detection Methods – Fault Recovery Methods – Redundancy-Based Fault Tolerance – Graceful Degradation.

UNIT V	RECONFIGURABLE SYSTEMS AND SURVEY OF NEUROMORPHIC PLATFORMS	9

Reconfigurable Neuromorphic Computing System – Reconfigurability in Neuromorphic Systems – Dynamic Reconfiguration – Network Topology Independence – Network Expandability – FPGA-Based Neuromorphic Design. Case Study: Real Hardware-Software Design of 3D-NoC-Based Neuromorphic System – 3D-IC Technology for Neuromorphic Computing – 3D Network-on-Chip Design. Survey of Neuromorphic Systems – IBM TrueNorth – Intel Loihi – SpiNNaker – BrainScaleS – Tianjic – Comparison of Neuromorphic Platforms – Future Directions.

TOTAL: 45 PERIODS

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

CO1: Explain the fundamental concepts of neuromorphic computing, biological brain architecture, and the need for brain-inspired computing. (K2)

CO2: Describe spiking neural network models, spike encoding schemes, and their digital and analog implementations. (K2)

CO3: Apply synaptic learning rules including STDP and Hebbian learning to spiking neural network architectures. (K3)

CO4: Analyze emerging memory technologies and crossbar array architectures for neuromorphic computing applications. (K4)

CO5: Evaluate communication network architectures and fault-tolerant design strategies for scalable neuromorphic systems. (K5)

CO6: Compare existing neuromorphic platforms and assess their suitability for different application requirements. (K5)

TEXT BOOKS:

1. Abderazek Ben Abdallah, Khanh N. Dang, "Neuromorphic Computing Principles and Organization", First Edition, Springer, 2022.

REFERENCES:

1. Khaled Salah Mohamed, "Neuromorphic Computing and Beyond: Parallel, Approximation, Near Memory, and Quantum", First Edition, Springer, 2020.
2. Christoph von der Malsburg, William A. Phillips, Wolf Singer (Eds.), "Dynamic Coordination in the Brain: From Neurons to Mind", MIT Press, 2010.
3. Giacomo Indiveri, Shih-Chii Liu, "Memory and Information Processing in Neuromorphic Systems", Proceedings of the IEEE, Vol. 103, No. 8, 2015.
4. Mike Davies et al., "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning", IEEE Micro, Vol. 38, No. 1, 2018.
5. Peter Dayan, L. F. Abbott, "Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems", MIT Press, 2001.

24AM934	GRAPH NEURAL NETWORKS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To understand fundamentals of graph theory.
- To study proof techniques related to various concepts in graphs.
- To apply graph theory to solve real-world problems.
- To analyze graphs and implement them.
- To explore modern applications of graph theory.

UNIT I	INTRODUCTION	9

Introduction - Graph Terminologies - Types of Graphs - Sub Graph- Multi Graph – Regular Graph - Isomorphism - Isomorphic Graphs - Euler graph - Hamiltonian Graph - Related Theorems.

UNIT II	TREES AND CONNECTIVITY	9

Trees -Properties- Distance and Centres - Types - Rooted Tree- Tree Enumeration Labeled Tree - Unlabeled Tree - Spanning Tree - Fundamental Circuits- Cut Sets - Properties - Fundamental Circuit and Cut-set- Connectivity- Separability -Related Theorems.

UNIT III	TRAVERSABILITY	9

Network Flows - Planar Graph - Representation - Detection - Dual Graph - Geometric and Combinatorial Dual - Related Theorems - Digraph - Properties - Euler Digraph.

UNIT IV	PLANARITY AND COLOURING	9

Matrix Representation - Adjacency matrix- Incidence matrix- Circuit matrix - Cut-set matrix - Path Matrix- Properties - Related Theorems - Correlations. Graph Coloring – Chromatic Polynomial - Chromatic Partitioning - Matching - Covering - Related Theorems.

UNIT V	PATTERN CLASSIFICIATION	9

Architecture-Biases and Threshold-Linear Separability-Data Representation-Hebb Net –Perceptron-Adaline-Pattern Association.

TOTAL:45 PERIODS

OUTCOMES:

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

CO1: Understand the basic concepts of graphs, and different types of graphs.
CO2: Illustrate the properties, theorems and be able to prove theorems.
CO3: Apply suitable graph model and algorithm for solving applications.
CO4: Understand the Various theorems of Matrix.
CO5: Understand the impact of data in Pattern Association.
CO6: Elaborate the concepts of Graph Theory.

TEXT BOOKS:

1. Narsingh Deo, "Graph Theory with Application to Engineering and Computer Science", Prentice-Hall of India Pvt. Ltd, 2003.
2. R. Foulds , "Graph Theory Applications", Springer ,2016.
3. Laurene V. Fausett, “Fundamentals of Neural Networks: Architectures, Algorithms and Applications”, Prentice-Hall, 1994.

REFERENCES:

1. Bondy, J. A. and Murty, U.S.R., " Graph Theory with Applications", North Holland Publication, 2008.
2. West, D. B., “Introduction to Graph Theory”, Pearson Education,2011.
3. John Clark , Derek Allan Holton, “A First Look at Graph Theory”, World Scientific Publishing Company, 1991.
4. Diestel, R, "Graph Theory", Springer,3rd Edition, 2006.

24AM935	EXPLAINABLE AND TRUSTWORTHY AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamentals and importance of Explainable Artificial Intelligence (XAI).
- To learn interpretable machine learning models and visualization techniques for model understanding.
- To explore advanced and post-hoc interpretability methods for explaining AI predictions.
- To study explainability techniques used in deep learning models and neural networks.
- To analyze real-world applications, challenges, and future directions of XAI systems

UNIT I	Foundations of Explainable AI	9

Introduction to Interpretability and Explainability- Fundamentals of interpretability, Need for explainable AI-Pre-Model Interpretability and Explainability, Data understanding and exploratory analysis, Feature relationships and preprocessing

UNIT II	Interpretable Models and Visualization	9

Model Visualization Techniques-Traditional Interpretable Algorithms-Linear models, Decision trees- Understanding model behavior through visualization

UNIT III	Advanced and Post-hoc Interpretability	9

Model Interpretability: Advances in Interpretable Machine Learning-Advanced interpretable techniques, Improvements over traditional methods-Post-hoc Interpretability and Explanations-Model-agnostic explanation techniques, Local explanations, Global explanations

UNIT IV	Explainable Deep Learning	9

Explainable Deep Learning-Interpreting neural networks-Deep learning explanation methods

UNIT V	Applications, Challenges and Future of XAI	9

Explainability in Time Series, NLP, and Computer Vision-Domain-specific explainability-Applications in: Time series forecasting, Natural language processing, Computer vision- XAI: Challenges and Future -Limitations of current methods

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the basic concepts, need, and preprocessing methods used in Explainable AI.

CO2: Apply interpretable models and visualization techniques to understand machine learning behavior.

CO3: Analyze advanced and post-hoc explanation methods for AI model interpretation.

CO4: Interpret deep learning models using suitable explainability techniques.

CO5: Evaluate XAI applications, limitations, and challenges in domains such NLP, computer vision, and time series..

CO6: Develop a mini project using XAI techniques to explain and evaluate the predictions of an AI model.

TEXT BOOK:

1. S. M. Lundberg, G. G. Erion, and S.-I. Lee, Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning. Cham, Switzerland: Springer, 2021

REFERENCES:

1. C. Molnar, Interpretable Machine Learning. 2nd ed., 2022.
2. A. B. Arrieta et al., Explainable AI: Foundations, Methodologies and Applications. Cham, Switzerland: Springer, 2022.

GENERATIVE AI

24AM929	LARGE LANGUAGE MODELS	L	T	P	C
		2	0	2	3

OBJECTIVES: The Course will enable learners to:

- Understand text preprocessing, tokenization, and vector representations used in NLP and LLMs.
- Learn the basic concepts of language models used in Large Language Models (LLMs).
- Understand Transformer architecture and Large Language Models for text generation tasks.
- Use pre-trained Hugging Face models for NLP applications and fine-tuning.
- Build LLM-powered applications using Prompt Engineering, RAG, and PEFT techniques.

UNIT I	FOUNDATIONS OF NLP AND EMBEDDINGS	6+6

Introduction to NLP – Words, Tokens, and Subword Tokenization – Unicode and Text Preprocessing – Byte Pair Encoding (BPE) – Corpora and Regular Expressions – Word Embeddings – Vector Semantics – Cosine Similarity – Word2Vec – Semantic Relationships in Embeddings – Bias and Evaluation of Embeddings.

List of Exercises / Experiments:

1.Perform text preprocessing using NLTK/spaCy including tokenization, stop-word removal, and lemmatization.

2.Train Word2Vec embeddings using Gensim and visualize semantic similarity between words.

UNIT II	LANGUAGE MODELS	6+6

Introduction to Language Models – Statistical Language Models – N-gram Language Models – Probability in NLP – Markov Assumptions – Smoothing Techniques – Neural Language Model Concepts – Continuous Bag of Words (CBOW) – Skip-Gram Model – Perplexity and Evaluation Metrics – Tokenization Strategies – Contextual Representations – Text Corpora and Vocabulary Building – Language Modeling Applications in NLP.

List of Exercises / Experiments:

3. Implement an N-gram language model and calculate perplexity for sample text.

4. Train CBOW and Skip-Gram models using Word2Vec and compare semantic similarity results.

UNIT III	TRANSFORMERS AND LARGE LANGUAGE MODELS	6+6

Introduction to Large Language Models – Auto-Regressive Language Modeling – Prompting Techniques – Text Generation and Sampling – Transformer Architecture – Self-Attention – Multi-Head Attention – Positional Encoding – Decoder-Only Architectures – Training and Evaluation of LLMs – Ethical and Safety Issues.

List of Exercises / Experiments:

5.Implement scaled dot-product attention and visualize attention weights using PyTorch.

6.Build a mini GPT-style Transformer model for text generation.

UNIT IV	PRE-TRAINED AND MULTIMODAL MODELS	6+6

Masked Language Models – BERT and Contextual Embeddings – Fine-Tuning for Classification and Named Entity Recognition – Hugging Face Transformers and Trainer API – Encoder–Decoder Translation Models – Beam Search – Diffusion Models – Stable Diffusion – Text-to-Image Generation – Multimodal AI Applications

List of Exercises / Experiments:

7. Fine-tune DistilBERT for sentiment classification using HuggingFace Transformers.

8. Build a Named Entity Recognition (NER) model using BERT and HuggingFace Trainer API.

UNIT V	LLM APPLICATIONS AND OPTIMIZATION	6+6

Prompt Engineering – Zero-shot and Few-shot Prompting – Chain-of-Thought Prompting – Retrieval-Augmented Generation (RAG) – Dense Vector Retrieval – Vector Databases – LoRA and QLoRA – Quantization Techniques – Parameter-Efficient Fine-Tuning (PEFT) – HuggingFace

Pipelines and Open-Source LLM Applications.

List of Exercises / Experiments:

9.Build a Retrieval-Augmented Generation (RAG) pipeline using FAISS/Chroma and Hugging Face embeddings.

10. Fine-tune an open-source LLM using LoRA or QLoRA on a small custom dataset.

TOTAL: 30 + 30 = 60 PERIODS

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

CO1: Explain NLP preprocessing, tokenization, and embedding techniques used in language models.

CO2: Explain statistical and neural language modeling techniques used as foundations for Large Language Models**.**

CO3: Analyze Transformer architecture and implement Large Language Models for text generation tasks.

CO4: Use and fine-tune pre-trained HuggingFace models for NLP applications.

CO5: Develop LLM-powered systems using Prompt Engineering, RAG, and PEFT techniques.

CO6: Evaluate ethical issues such as bias, hallucination, and privacy in Large Language Models

TEXT BOOKS:

1. Daniel Jurafsky and James H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models", 3rd Edition, Stanford University, 2026. [Primary reference – Volume I: Large Language Models, Chapters 1–13]. Available free online: https://web.stanford.edu/~jurafsky/slp3/

REFERENCES:

1. Tong Xiao and Jingbo Zhu, "Foundations of Large Language Models", NLP Lab, Northeastern University and NiuTrans Research, 2025 (Creative Commons Licensed). Available free online: https://arxiv.org/abs/2501.09223
2. Sebastian Raschka, "Build a Large Language Model (From Scratch)", 1st Edition, Manning Publications, 2024. Companion code (open source): https://github.com/rasbt/LLMs-from-scratch
3. Tanmoy Chakraborty, "Introduction to Large Language Models", 1st Edition, Wiley India, 2025. ISBN: 9789363864740.
4. Jacob Eisenstein, "Natural Language Processing", 1st Edition, The MIT Press, 2019.
5. HuggingFace Team, "HuggingFace NLP Course" and "Transformers Documentation". Freely available at https://huggingface.co/learn/nlp-course

24AM946	MULTIMODAL AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamentals, evolution, and significance of multimodal artificial intelligence systems integrating text, image, audio, and generative modalities.
- To analyze the architecture and working principles of multimodal chatbot systems and their interaction mechanisms across different AI modalities.
- To explore multimodal data processing techniques including data integration, alignment, fusion, and preprocessing methods for intelligent applications.
- To examine real-world applications, implementation challenges, deployment strategies, and performance optimization techniques in multimodal AI systems.
- To evaluate ethical considerations, responsible AI practices, and emerging trends such as agentic AI and embodied intelligence in multimodal AI development.

UNIT I	INTRODUCTION TO MULTIMODAL AI	9

Introduction to Multimodal AI-What is multimodal AI, Evolution of AI systems, Importance of combining modalities-Understanding AI Modalities-Text (Natural Language Processing), Image (Computer Vision), Audio (Speech Recognition), Generative capabilities.

UNIT II	Multimodal Chatbot Architecture	9

How Multimodal Chatbots Work-Architecture of multimodal systems, Interaction between modalities, Chatbot design principles-Data Integration and Processing-Multimodal data handling, Data alignment and fusion, Data pre-processing techniques.

UNIT III	Applications of Multimodal AI	9

Real-World Applications-Healthcare applications, Education systems, Manufacturing and industry, Creative and media applications

UNIT IV	System Design and Challenges	9

Implementation Challenges-Data integration issues, Privacy and security concerns, Scalability challenges, Performance optimization-Building Multimodal AI Systems- Frameworks and development strategies, Deployment considerations, Evaluation and testing

UNIT V	Ethics and Future Trends	9

Ethics and Best Practices- Responsible AI principles, Bias and fairness, Ethical deployment-Future Trends in Multimodal AI-Agentic AI systems, Embodied intelligence, Emerging technologies-Implementation Guidance-Adoption strategies-Practical recommendations

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the concepts, evolution, and core components of multimodal AI Systems and their modalities.

CO2: Design and analyze multimodal chatbot architectures with appropriate integration and interaction mechanisms

CO3: Apply multimodal data preprocessing, alignment, and fusion techniques for AI-based applications.

CO4: Analyze the role of multimodal AI in domains such as healthcare, education, manufacturing, and creative industries.

CO5: Identify challenges related to scalability, privacy, security, and performance while developing multimodal AI systems.

CO6: Evaluate ethical issues, responsible AI practices, and future advancements including agentic AI and embodied intelligence for sustainable AI deployment.

TEXT BOOK:

1. I. Sterling, Multimodal AI: How Chatbots Learn to See, Hear, and Create. Forge Prod, 2025.

REFERENCES:

1. L. Lin and Y. Liu, Multimodal Large Models: A New Paradigm of Artificial Intelligence. Singapore: Springer, 2026

2. A. Singh and K. K. Singh, Eds., Multimodal Generative AI. Singapore: Springer, 2025.

24AM952	VECTOR DATABASES	L	T	P	C
		2	0	2	3

OBJECTIVES:

The Course will enable learners to:

1. Understand the need for vector databases and how they differ from traditional SQL and NoSQL databases.
2. Learn the fundamentals of vector embeddings including Word2Vec, Doc2Vec, and Transformer-based embeddings.
3. Explore similarity search techniques, distance metrics, and approximate nearest neighbor algorithms.
4. Gain hands-on experience with vector-enabled databases such as SQLite-VSS and PostgreSQL with pgvector.
5. Build Retrieval-Augmented Generation (RAG) applications using vector databases and large language models.

UNIT I	INTRODUCTION TO VECTOR DATABASES	6+6

Introduction to Vector Databases – Why Do You Need Vector Databases? A New Data Type: Vector. Similarity Search. What's Different About the Vector Type? Where Do You Use Vector Databases? SQL Versus Vector Databases – The Foundation of Business Math: Accounting Arithmetic. Vector Representation in a Relational Database Management System. The Need for Vector-Specific Capabilities. NoSQL Versus Vector Databases – NoSQL Databases and Vector Storage. Limitations of Vector Extensions in NoSQL Databases. When to Choose NoSQL with Vector Extensions. Hybrid Approaches: Combining Structured and Vector Data – The Need for Both Vector Data and Metadata. Limitations of Pure Vector Storage. Hybrid Database Architecture. Benefits of the Hybrid Approach.

List of Exercise/Experiments:

1. Explore and compare SQL, NoSQL, and vector databases by performing keyword search vs. similarity search on a sample text dataset.
2. Install and set up a Python environment with required libraries (NumPy, scikit-learn) and perform basic vector arithmetic operations.
3. Implement cosine similarity and Euclidean distance calculations on sample vectors using Python.

UNIT II	EMBEDDINGS	6+6

Embeddings – Understanding Vector Embeddings: Why We Need Them. Word2Vec: The Breakthrough That Changed Everything. Doc2Vec: From Words to Documents. From Embeddings to Modern Language Models: The Transformer Connection – Encoder-Only Transformers (BERT and Its Variants). Decoder-Only Transformers (GPT Family). Encoder-Decoder Transformers (T5, BART). Embedding Models: The Specialized Vector Generators – Distinction from Traditional Models. Role in Modern LLM Applications. Practical Applications and Use Cases. The Sentence-Transformers Library. The Embedding Layer: The Gateway to Zero-Shot Learning. Vector Arithmetic with Word2Vec.

List of Exercise/Experiments:

1. Generate word embeddings using a pre-trained Word2Vec model and perform vector arithmetic (e.g., king – man + woman = queen).
2. Generate sentence-level embeddings using the Sentence-Transformers library and compute pairwise similarity scores.
3. Compare embeddings from different models (Word2Vec, BERT, Sentence-Transformers) on a given text corpus and visualize using t-SNE.

UNIT III	SQLITE VECTOR SIMILARITY SEARCH	6+6

SQLite Vector Similarity Search (SQLite-VSS) – Understanding the SQLite Vector Similarity Search Extension: Core Capabilities. Architecture Overview. Limitations. Setting Up the

Development Environment – Installing Dependencies. Verifying the Installation. Operational Pragmas. Designing the Database Schema – Schema Requirements. Table Definitions. Schema Design Decisions. Content Extraction and Preprocessing – Text Cleaning Pipeline. Quality Filtering. Generating and Storing Embeddings – Embedding Generator. Database Storage. Batch Processing Pipeline. Building the Vector Index – Understanding VSS Indexing. Index Management. Implementing Semantic Search – Search Result Container. Search Engine.

List of Exercise/Experiments:
1. Set up SQLite with the VSS extension, create a schema, and store text embeddings for a sample document collection.
2. Build a basic semantic search engine using SQLite-VSS that retrieves the top-K similar documents for a user query.
3. Implement a text cleaning and chunking pipeline and evaluate the impact of chunk size on search quality.

UNIT IV	**POSTGRESQL AND HYBRID SEARCH**	**6+6**

Semantic Search with SQLite3 – Putting It All Together: Workflow Example. Example Output. Extension: Incremental Indexing. PostgreSQL as a Vector Database with pgvector – Database Foundation with Vector Support: Setting Up the Vector-Enabled Database. Schema Design for RAG. Creating Search Indexes. Text Processing and Embedding Generation – Embedding Model Management. Intelligent Text Chunking. Storing Content with Embeddings. Hybrid Search Implementation – Hybrid Search Algorithm. Semantic Search Component. Keyword Search Component. Score Fusion and Ranking. Cloud-Managed Vector Databases – Overview of Pinecone, Weaviate, and ChromaDB. Pinecone Architecture: Serverless Indexes, Namespaces, and Metadata Filtering. Comparing Self-Hosted vs. Cloud-Managed Vector Databases.

List of Exercise/Experiments:
1. Install PostgreSQL with pgvector extension, design a vector-enabled schema, and load a dataset with embeddings.
2. Implement hybrid search (combining semantic and keyword search) using pgvector and compare results with pure semantic search.
3. Using Pinecone free tier: Create a serverless index, upsert text embeddings with metadata, perform similarity search with metadata filtering, and compare query latency and results with pgvector on the same dataset.

UNIT V	**RAG PIPELINE AND APPLICATIONS**	**6+6**

LLM Integration and RAG Pipeline – LLM Integration with Ollama: Ollama API Client. Health Check Function. The RAG Pipeline – Context Formatting. Question-Answering Pipeline. Demonstration and Testing – Sample Data Loading. Main Demonstration Function. Interactive Q&A Interface. Quick Testing Utility. Next Steps: Extending the System – Performance Optimizations. Production Considerations. Advanced RAG Patterns.

List of Exercise/Experiments:
1. Build an end-to-end RAG pipeline that retrieves relevant context from a vector database and generates answers using a local LLM.
2. Evaluate the RAG system by comparing responses with and without retrieval augmentation on a set of benchmark questions.
3. Mini-Project: Develop a domain-specific intelligent Q&A system (e.g., university FAQ bot or medical knowledge assistant) using vector databases and RAG with a locally hosted LLM**.**

TOTAL: 30+30 = 60 PERIODS

OUTCOMES:

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

CO1: Explain the need for vector databases and differentiate them from traditional SQL and NoSQL databases.

CO2: Generate vector embeddings for text data using Word2Vec, Sentence-Transformers, and Transformer-based models.

CO3: Build and manage vector indexes using SQLite-VSS, PostgreSQL with pgvector, and Pinecone for efficient similarity search.

CO4: Analyze the performance of different similarity metrics, indexing strategies, and search algorithms for vector retrieval.

CO5: Design hybrid search systems that combine semantic search and keyword search with score fusion and ranking.

CO6: Develop a Retrieval-Augmented Generation (RAG) application integrating vector databases with large language models.

TEXT BOOKS:

1. Nitin Borwankar, "Vector Databases: A Practical Introduction", O'Reilly Media, March 2026. (ISBN: 978-1098177591)

REFERENCES:

1. Sebastian Witalec, "Vector Databases: From Embeddings to Applications" – DeepLearning.AI Short Course (Weaviate), 2023. [Available online: https://learn.deeplearning.ai]

2. Valentina Alto, "Building LLM Powered Applications", Packt Publishing, May 2024. (ISBN: 978-1835462317)

3. James Jie Pan, Jianguo Wang, and Guoliang Li, "Vector Database Management Techniques and Systems", ACM SIGMOD 2024 Tutorial. [Available online: https://www.cs.purdue.edu]

4. IBM Skills Network, "Vector Database Fundamentals Specialization" – Coursera, 2024. [Available online: https://www.coursera.org]

5. Scrimba / Coursera, "Retrieval-Augmented Generation (RAG) with Embeddings & Vector Databases", 2024. [Available online: https://www.coursera.org]

24AM953	PROMPT ENGINEERING AND LLM OPTIMIZATION	L	T	P	C
		2	0	2	3

OBJECTIVES: The Course will enable learners to:

- Understand the architecture and inner workings of large language models including tokenization, auto-regressive generation, and the transformer architecture.
- Learn how reinforcement learning from human feedback (RLHF) shapes LLM behavior and the differences between instruct and chat models.
- Develop skills in designing LLM applications by sourcing context, assembling prompts, and managing token limits effectively.
- Explore techniques for taming model outputs using temperature, logprobs, and steering strategies.
- Build conversational agents with tool integration and design multi-step LLM workflows and pipelines.
- Evaluate LLM applications using offline and online testing frameworks, and understand emerging trends in multimodality and ethics.

UNIT I	INTRODUCTION TO PROMPT ENGINEERING AND UNDERSTANDING LLMs (Ch 1–2)	6+6

Introduction to Prompt Engineering: LLMs Are Magic – Language Models: How Did We Get Here? – Early Language Models – GPT Enters the Scene – Prompt Engineering.
Understanding LLMs: What Are LLMs? – Completing a Document – Human Thought Versus LLM Processing – Hallucinations – How LLMs See the World – LLMs Use Deterministic Tokenizers – LLMs Can't Slow Down and Examine Letters – LLMs See Text Differently – Counting Tokens – One Token at a Time – Auto-Regressive Models – Patterns and Repetitions – Temperature and Probabilities – The Transformer Architecture.

List of Exercises/Experiments:

1. Set up a Python environment with Ollama and an open-source LLM (e.g., Llama or Mistral). Write basic prompts and observe how the model completes documents in different styles.
2. Experiment with temperature and probability settings: run the same prompt at temperature 0, 0.5, and 1.0, and document the variation in outputs.
3. Explore tokenization by tokenizing the same text using different open-source tokenizers and compare token counts and boundary differences.

UNIT II	RLHF, CHAT MODELS, AND DESIGNING LLM APPLICATIONS (Ch 3–4)	6+6

Reinforcement Learning from Human Feedback: The Process of Building an RLHF Model – Keeping LLMs Honest – Avoiding Idiosyncratic Behavior – RLHF Packs a Lot of Bang for the Buck – Beware of the Alignment Tax.
Moving from Instruct to Chat: Instruct Models – Chat Models – The Changing API – Chat Completion API – Comparing Chat with Completion Models.
Designing an LLM Application: From Problem to Application – The LLM Application Design Process – Choosing the Right Model – Defining the Task – Iteration and Refinement.

List of Exercises/Experiments:

1. Compare the outputs of a base model and an instruction-tuned model (e.g., Llama base vs. Llama-Instruct) for the same prompts and document differences in helpfulness and safety.
2. Interact with an open-source chat model using both completion-style and chat-style APIs. Observe how system, user, and assistant roles affect the output.
3. Design a simple LLM application: define a problem statement, choose a model, write an initial prompt, and iteratively refine it across three versions with documented improvements.

UNIT III	CONTEXT, ASSEMBLING THE PROMPT, AND TAMING THE MODEL (Ch 5–7)	6+6

Context: Sourcing Context Information – Ranking Context by Importance – Selecting Relevant Context for the Task – Few-Shot Learning – Chain-of-Thought Prompting.
Assembling the Prompt: Markdown Formatting – Snippet Prioritization – Avoiding Token Limits – Prompt Templates – Organizing Prompt Elements – Structuring for High-Quality Completions.
Taming the Model: Logprobs – Temperature Tuning – Mitigating Repetition – Steering Completions – Handling Off-Topic Tangents – Controlling Output Format and Length.

List of Exercises/Experiments:

1. Implement few-shot learning and chain-of-thought prompting for a classification task (e.g., sentiment analysis). Compare accuracy against zero-shot prompting.
2. Build prompt templates with different context arrangements (context-first vs. instruction-first) and measure output quality on a set of 10 test inputs.
3. Experiment with logprobs and temperature tuning to control output determinism. Create a structured output task (e.g., JSON extraction) and refine prompts to achieve consistent formatting.

UNIT IV	CONVERSATIONAL AGENCY AND LLM WORKFLOWS (Ch 8–9)	6+6

Conversational Agency: Designing Stateful Chatbots – Tool Integration – APIs and Databases – ReAct: Reasoning and Acting – Building Agents with Memory – Managing Conversation History.
LLM Workflows: Orchestrating Multi-Step Processes – Pipelines and Workflow Design – Task Decomposition – Chaining LLM Inferences – Error Handling in Workflows – Practical Workflow Examples.

List of Exercises/Experiments:

1. Build a stateful chatbot using an open-source LLM that maintains conversation history across multiple turns. Test with a customer support scenario.
2. Implement a ReAct-style agent that uses a calculator tool and a dictionary lookup tool to answer multi-step questions requiring reasoning and action.
3. Design a multi-step LLM workflow pipeline: take a user's topic, generate an outline, expand each section, and produce a final formatted document – all orchestrated in Python.

UNIT V	EVALUATING LLM APPLICATIONS AND LOOKING AHEAD (Ch 10–11)	6+6

Evaluating LLM Applications: Offline Evaluation – Building Example Suites – Code Correctness Testing – Online Evaluation – A/B Testing – User Engagement Metrics – Direct Feedback Data – Tracking Incidental Metrics – Using Evaluation Data for Fine-Tuning.
Looking Ahead: Multimodality – User Interfaces for LLM Applications – Ethical Considerations – Core Principles for Effective Prompt Engineering – Future Trends in LLM Applications.

List of Exercises/Experiments:

1. Build an offline evaluation suite for a text summarization task: create a gold standard dataset of 15 examples, run prompts against them, and score outputs using a rubric.
2. Design an A/B test for two prompt strategies on the same task. Run both on 20+ inputs, collect metrics, and write a short analysis report on which performs better and why.
3. Mini-Project: Design and build an end-to-end LLM-powered application (e.g., study planner, code reviewer, or interview preparation assistant) with a Streamlit/Gradio interface, incorporating prompt templates, conversation memory, and an evaluation component. Present with a demo and report.

TOTAL: 30+30 = 60 PERIODS

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

CO1: Explain the architecture of large language models including tokenization, auto-regressive generation, hallucinations, and the transformer mechanism. *[K2 – Understand]*

CO2: Describe the RLHF process, the alignment tax, and the differences between instruct and chat models. *[K2 – Understand]*

CO3: Apply context sourcing, few-shot learning, chain-of-thought prompting, and prompt assembly techniques to build effective prompts. *[K3 – Apply]*

CO4: Analyze and compare prompt strategies using logprobs, temperature tuning, and structured output techniques to optimize LLM responses. *[K4 – Analyze]*

CO5: Design conversational agents with tool integration and orchestrate multi-step LLM workflows for real-world applications. *[K6 – Create]*

CO6: Evaluate LLM applications using offline example suites, online A/B testing, and user engagement metrics with appropriate evaluation frameworks. *[K5 – Evaluate]*

TEXT BOOKS:

1. John Berryman and Albert Ziegler, "Prompt Engineering for LLMs: The Art and Science of Building Large Language Model–Based Applications", O'Reilly Media, 2025. ISBN: 9781098156138.

REFERENCES:

1. James Phoenix and Mike Taylor, "Prompt Engineering for Generative AI: Future-Proof Inputs for Reliable AI Outputs", O'Reilly Media, 2024. ISBN: 9781098153434.
2. DAIR.AI, "Prompt Engineering Guide". Freely available at https://www.promptingguide.ai
3. NPTEL Course: "Introduction to Large Language Models (LLMs)" by Prof. Tanmoy Chakraborty, IIT Delhi. Available at https://onlinecourses.nptel.ac.in/noc25_cs45
4. Hugging Face Documentation: Transformers and PEFT. Freely available at https://huggingface.co/docs
5. Anthropic, "Prompt Engineering Documentation". Freely available at https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering

24AM954	ETHICAL AND RESPONSIBLE GENERATIVE AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the fundamentals and evolution of Generative AI models.
- To analyze architectures such as GANs, VAEs, and Transformers.
- To explore ethical challenges, bias, and risks in Generative AI systems.
- To study responsible AI frameworks, safety, and trustworthiness.
- To examine governance, policy, and societal implications of Generative AI.

UNIT I	INTRODUCTION TO GENERATIVE AI	9

Introduction – Definition and scope of Generative AI – Evolution from traditional AI to Generative models – Overview of deep learning and probabilistic foundations– Generative vs Discriminative models – Applications of Generative AI (text, image, and audio) – Introduction to large language models – Challenges and future directions.

UNIT II	GENERATIVE MODELS AND TECHNIQUES	9

Generative Adversarial Networks (GANs) – Variational Autoencoders (VAEs) – Transformer architectures – Latent space representation– Evaluation metrics (FID, BLEU, ROUGE) – Diffusion models – Training techniques and evaluation metrics – Prompt engineering basics – Fine-tuning and transfer learning – Case studies and research trends.

UNIT III	ETHICAL ISSUES IN GENERATIVE AI	9

Bias and fairness in generative models – Hallucination and misinformation – Deepfakes and synthetic media risks – Explainability challenges in generative models– Privacy concerns and data misuse – Environmental impact of large models – Human-AI interaction risks – Case studies in ethical failures.

UNIT IV	RESPONSIBLE AI FRAMEWORKS FOR GENERATIVE AI	9

Principles of Responsible AI – Fairness, Accountability, Transparency (FAT) –Model auditing and validation–AI risk assessment framework– Explainability in generative models – Safety alignment and red-teaming – Secure AI systems and adversarial threats – Responsible deployment lifecycle – Industry frameworks and best practices.

UNIT V	GOVERNANCE, POLICY AND FUTURE OF GENERATIVE AI	9

AI governance frameworks – Global AI regulations (EU AI Act, OECD principles) –Data governance and data protection laws (GDPR basics) – Policy challenges in Generative AI – Intellectual property and copyright issues – AI for social good – Risks of superintelligence – Future of Generative AI ecosystems – Responsible innovation and global perspectives.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain fundamentals and evolution of Generative AI models.
CO2: Apply generative techniques such as GANs, VAEs, and Transformers.
CO3: Analyze ethical issues including bias, privacy, and misinformation.
CO4: Evaluate responsible AI frameworks and safe deployment strategies.
CO5: Assess governance policies and societal impacts of Generative AI.
CO6: Examine future challenges and emerging trends in Generative AI.

TEXT BOOK:

1. David Foster, “Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play”, O’Reilly Media, 2nd Edition, 2022, ISBN: 978-1098134187.
2. Virginia Dignum, “Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way”, Springer Nature, 2019, ISBN: 978-3030303566

REFERENCES:

1. Hannah Fry, “Hello World: How to Be Human in the Age of the Machine”, W. W. Norton & Company, 2019, ISBN: 978-0393357369.

2. Jonathan Birch, “The Edge of Sentience: Risk and Precaution in Humans, Other Animals, and AI”, Oxford University Press, 2024, ISBN: 978-0198871718
3. Jeff Sebo, “The Moral Circle: Who Matters, What Matters, and Why”, W. W. Norton & Company, 2023, ISBN: 978-1324052520**.**
4. Kate Crawford, “Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence”, Yale University Press, 2021, ISBN: 978-0300209576.

24AM955	RAG AND KNOWLEDGE SYSTEMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Understand the fundamentals of Large Language Models (LLMs) and the need for Retrieval-Augmented Generation (RAG).
- Learn the design and development of RAG-based applications using modern frameworks and tools.
- Analyze techniques for evaluation and optimization of RAG systems.
- Explore methods for building scalable, production-ready, and observable AI systems.
- Examine governance, security, and advanced approaches such as multimodal RAG and knowledge graphs.

UNIT I	INTRODUCTION TO RAG	9

Overview of Large Language Models - Building Industry LLM Applications - Introduction to Retrieval-Augmented Generation (RAG) - Architecture of RAG systems - Industry use cases of RAG - Overview of Haystack framework -
Building industry-scale LLM applications - Hands-on: Build a basic RAG application using Haystack

UNIT II	EVALUATING AND OPTIMIZING RAG SYSTEMS	9

Components of a RAG application - RAG Evaluation - Pipeline Optimizations - Optimizing Chunking - Optimizing Embeddings and Storage - Basic Pipeline for Document QA - Evaluating the Pipeline - Optimizing Retrieval - Optimizing Generation

UNIT III	SCALABLE AND PRODUCTION-READY RAG SYSTEMS	9

Transition from prototype to production - Production-Ready RAG - Deploying LLMs - Deploying Embedding Models - Databases in Production - RAG in Production with Haystack - Haystack RAG application architecture - Haystack Pipeline Code - Running Experiments in Production -

UNIT IV	OBSERVABLE AI AND MONITORING	9

Observable AI - Data and Concept Drifts - Logging and Tracing - GenAI Monitoring – Quality – Security - Security with Haystack - Latency and Costs - GenAI Monitoring with Haystack

UNIT V	GOVERNANCE, ADVANCED RAG, AND KNOWLEDGE SYSTEMS	9

Cost Management - Data and Privacy - Security and Safety - Advanced RAG and Keeping Pace with AI Developments - AI Agents - Multimodal RAG - Knowledge Graphs for RAG - SQL RAG

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the architecture and working principles of RAG systems and LLM-based applications.
CO2: Develop basic RAG applications using appropriate frameworks and tools.
CO3: Analyze the performance of RAG systems using suitable evaluation metrics.
CO4: Optimize RAG pipelines for improved retrieval and generation efficiency..
CO5: Evaluate scalability, monitoring, and governance aspects of production-level RAG systems.
CO6: Design advanced RAG solutions integrating knowledge graphs, multimodal data, or agent-based approaches..

TEXT BOOK:

1. Skanda Vivek. Generation in Production with Haystack, O'Reilly, 2025

REFERENCES:

1. Ranajoy Bose. Mastering Retrieval-Augmented Generation: Advanced Techniques and Production-Ready Solutions for Enterprise AI, Springer, 2025

24AM956	Generative AI for Software Engineering	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Understand the transformative impact of Generative AI on the software development lifecycle.
- Learn prompt engineering techniques specific to code generation, debugging, and documentation tasks.
- Explore AI-powered tools for code generation, code review, and testing across the SDLC.
- Apply Generative AI for software design, UI/UX development, and DevOps automation.
- Evaluate the challenges, risks, and best practices of integrating AI into software engineering workflows.

UNIT I	GENERATIVE AI IN THE SOFTWARE DEVELOPMENT LIFECYCLE	9

Industry Study on Developer Productivity with Generative AI – Transforming Software Development with Generative AI in the SDLC – Planning and Task Management – Coding and Development – Testing and Quality Assurance – CI/CD and AI-Powered Automation. Generative AI for Specific Programming Tasks – Code Generation – Code Completion – Bug Detection and Debugging – Code Refactoring and Optimization – Code Documentation – Automated Test Case – Code Reviews – Learning and Teaching Programming. End-to-End AI Integration in the SDLC – Challenges and Tradeoffs in AI Integration – Balancing AI Model Updates and Infrastructure Costs – Maintaining Developer Control Over AI-Generated Code – Ensuring Security and Compliance in AI-Generated Code.

UNIT II	PROMPT ENGINEERING FOR SOFTWARE DEVELOPMENT	9

Prompt Engineering for Software Developers – Why Prompt Engineering? – Anatomy of a Prompt – Top 5 Reasons to Learn Prompt Engineering. Prompt Techniques – Zero-Shot Prompting – Few-Shot Prompting – Chain of Thought (CoT) Prompting – Self-Consistency – ReAct: Reasoning and Acting Prompting Technique – Retrieval Augmented Generation (RAG). Prompt Use Cases for the Software Development Lifecycle (SDLC) – Code Debugging – Software Design Documentation – UML Diagram Generation – Test Case Generation – Deployment Script Generation. Prompt Management Cycle and Best Practices – Prompt Design – Prompt Testing and Evaluation – Prompt Refinement – Prompt Versioning and Cataloging – Prompt Deployment – Monitoring and Iteration – Prompt Engineering Tools.

UNIT III	AI-POWERED CODE GENERATION AND CODE REVIEW TOOLS	9

Types of Code-Generation Tools – Use Cases. Browser-Based Tools – ChatGPT – Google Gemini. IDE-Based Tools – GitHub Copilot – Cursor – Windsurf. Tool Comparison – ChatGPT – Google Gemini – GitHub Copilot – Cursor – Windsurf. Types of AI Code-Review Tools – Use Cases – Keeping the Human Review – Evaluation Process – Codacy – DeepCode (by Snyk) – CodeRabbit – Tool Comparison. Types of AI Tools for Design and Frontend Development – Pros and Cons of Using AI Tools for UI/UX – Use Cases for UI/UX AI Tools – Evaluation Process – UI Tools – Uizard – Bolt.new – Lovable – UX Tools – QoQo.ai – Research Studio – Tool Comparison.

UNIT IV	AI-POWERED TESTING, DOCUMENTATION, AND DevOps	9

Types of AI Testing Tools – Use Cases – The Need for Human Testers – Evaluation Process – Katalon Studio – testRigor – Tool Comparison. Types of AI Documentation Tools – Use Cases – Evaluation Process – Mintlify – Swimm – Tool Comparison. Types of AI DevOps Tools – Use Cases – Evaluation Process – Harness – Kubiya – Tool Comparison. Types of AI Security Tools – Use Cases – Evaluation Process – Snyk – Socket – Tool Comparison.

UNIT V	AI-POWERED PROJECT MANAGEMENT, EVALUATION, AND FUTURE	9

	OUTLOOK	
Types of AI Project-Management Tools – Use Cases – Evaluation Process – Linear – Shortcut – Tool Comparison. Key Metrics and KPIs for Measuring AI Impact – Evaluating AI-Generated Code Quality – Measuring Developer Productivity Gains – Benchmarks for AI Coding Assistants – HumanEval – MBPP – SWE-bench. Next Steps: Sustaining and Expanding AI Integration – Pilot AI Tools in Targeted Development Stages – Continuous Model Training and Feedback Loops – Establish Success Metrics and Iterate. The Future Outlook – Agentic Coding and Autonomous Software Development – Vibe Coding – Multi-agent Systems for Software Engineering – Real-World Impacts on Productivity and Satisfaction – Common Pitfalls and Risks – Perceived Job Impacts.		
TOTAL: 45 PERIODS		

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

CO1: Explain how Generative AI transforms each stage of the software development lifecycle. (K2)

CO2: Apply prompt engineering techniques for code generation, debugging, testing, and documentation tasks. (K3)

CO3: Analyze and compare AI-powered code generation, review, and testing tools for given development scenarios. (K4)

CO4: Design AI-integrated workflows for testing, DevOps automation, and security analysis in software projects. (K5)

CO5: Evaluate the quality of AI-generated code using benchmarks such as HumanEval, MBPP, and SWE-bench. (K5)

CO6: Assess the risks, limitations, and ethical considerations of deploying AI tools in professional software development. (K5)

TEXT BOOKS:

1. Albert Ziegler, Eirini Kalliamvakou, Ryan J. Salva, and Rizel Scarlett, “Generative AI for Software Development,” O’Reilly Media, 2025.

REFERENCES:

1. Saurabh Shrivastava, Kamal Arora, Ashutosh Dubey, Dhiraj Thakur, and Sanjeet Sahay, “Generative AI for Software Developers,” Packt Publishing, 2025.
2. Eran Yahav and Brent Laster, “The AI-Enabled SDLC,” O’Reilly Media, 2025.
3. Chip Huyen, “AI Engineering: Building Applications with Foundation Models,” O’Reilly Media, 2025.
4. Foutse Khomh et al., “Generative AI for Effective Software Development,” Springer, 2024.
5. Mark Chen et al., “Evaluating Large Language Models Trained on Code (Codex / HumanEval),” arXiv: 2107.03374, 2021.

24AM957	GEN AI SYSTEMS ENGINEERING AND LLMOps	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Understand the evolution from traditional MLOps to LLMOps and the core concepts of LLM lifecycle management.
- Learn data collection, preparation, and processing techniques specific to large language model operations.
- Explore model development workflows including feature engineering, foundation model selection, and fine-tuning.
- Apply inference optimization, model serving, and scalability strategies for deploying LLMs in production.
- Implement monitoring, continuous improvement, and governance frameworks for responsible LLM operations.

UNIT I	FOUNDATIONS OF LLMs AND LLMOps	9

Introduction to LLMs and LLMOps – The evolution of NLP and LLMs – The rise of machine learning in NLP – Deep learning revolution – The birth of LLMs – Current state and future directions. Traditional MLOps versus LLMOps – Stages in the MLOps life cycle – Specific challenges and methodologies in LLMOps. Trends in LLM integration – Integration of LLMs across industries – Current trends and examples of LLM applications. Core concepts of LLMOps – Key LLMOps-specific terminology – Model architecture – LLMOps workflow overview – Step-by-step overview – Real-world example. Reviewing LLMOps Components – Data collection and preparation – Model pre-training and fine-tuning – Governance and review – Inference, serving, and scalability – Monitoring – Continuous improvement.

UNIT II	DATA PROCESSING AND FEATURE ENGINEERING IN LLMOps	9

Processing Data in LLMOps Tools – Collecting data – Collecting structured data – Collecting semi-structured data – Collecting unstructured data. Transforming data – Defining core data attributes – Transforming data. Preparing data – Cleaning text data – Handling insufficient context – Transforming data for LLM consumption – Example Workflow in PySpark – Automating Spark Jobs. Developing Models via LLMOps – Creating features – Tokenizing annotations – Uniquely identifying tokens with attention masks – Storing features – Retrieving features – Selecting the foundation model – Choosing the LLM for your specific use case – Testing foundation LLMs – Addressing additional model concerns.

UNIT III	MODEL DEVELOPMENT, FINE-TUNING, AND REVIEW	9

Fine-tuning the foundation LLM – Tuning hyperparameters – Automating model development. LLMOps Review and Compliance – Evaluating LLM performance metrics offline – Evaluating binary, multi-class, and multi-label metrics – Evaluating perplexity, BLEU, and ROUGE – Evaluating reliability and robustness – Evaluating conversational flow. Securing and governing models with LLMOps – Managing OWASP risks in LLMs – Governance for LLMs. Ensuring legal and regulatory compliance – Operationalizing compliance and performance – Operationalizing performance – Security and governance – Legal and regulatory compliance – Validation of data and model licensing – Human review points.

UNIT IV	INFERENCE, SERVING, AND SCALABILITY	9

LLMOps Strategies for Inference, Serving, and Scalability – Operationalizing inference strategies in LLMOps – Decoding inference types – Real-time, batch, and interactive inference. Model pruning – Model quantization – Synergistic effects and considerations – Efficient hardware utilization – Trade-offs between inference speed and output quality. Optimizing model serving for performance – Comparing serverless, containerized, and microservices architectures – Leveraging the microservices architecture – Performance tuning – Serving up-to-date models – Rolling back failed deployments –

Increasing model reliability.

UNIT V	MONITORING, CONTINUOUS IMPROVEMENT, AND EMERGING TECHNOLOGIES	9

LLMOps Monitoring and Continuous Improvement – Monitoring LLMs fundamentals – Maintaining consistent performance – Compliance and security – Resource optimization – Fostering trust and reliability. Monitoring metrics and parameters for LLMs – Monitoring tools and technologies – Cloud-based platforms – Custom solutions – Key metrics to monitor. Learning from human feedback – Collecting and integrating feedback – Challenges in integrating feedback – Solutions for effective feedback integration – Impacts of human feedback. Incorporating continuous improvement – Key principles of continuous improvement in LLMOps – Integration of automation tools for seamless improvement cycles. The Future of LLMOps and Emerging Technologies – Advancements in model architectures – Scaling models – Automated Machine Learning (AutoML) – Federated learning – Edge computing – AI and IoT convergence – Responsible AI – Preparing for next-generation LLMs.

TOTAL: 45 PERIODS

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

CO1: Explain the evolution from MLOps to LLMOps and describe the key stages in the LLM lifecycle. (K2)

CO2: Apply data collection, preparation, and feature engineering techniques for LLM model development workflows. (K3)

CO3: Evaluate LLM performance using offline metrics such as perplexity, BLEU, ROUGE, and robustness measures. (K5)

CO4: Design inference optimization strategies using model pruning, quantization, and containerized serving architectures. (K5)

CO5: Develop monitoring dashboards and continuous improvement pipelines for deployed LLM systems. (K5)

CO6: Analyze governance, security, and regulatory compliance requirements for responsible LLM operations. (K4)

TEXT BOOKS:

1. Ryan Doan, “Essential Guide to LLMOps: Implementing Effective Strategies for Large Language Models in Deployment and Continuous Improvement,” Packt Publishing, 2024.

REFERENCES:

1. Abi Aryan, “LLMOps: Managing Large Language Models in Production,” O’Reilly Media, 2025.
2. Chip Huyen, “AI Engineering: Building Applications with Foundation Models,” O’Reilly Media, 2025.
3. Andriy Burkov, “Machine Learning Engineering,” True Positive Inc., 2020.
4. Hannes Hapke and Catherine Nelson, “Building Machine Learning Pipelines: Automating Model Life Cycles with TensorFlow,” O’Reilly Media, 2020.
5. Google Cloud, “MLOps: Continuous Delivery and Automation Pipelines in Machine Learning,” 2023. [Available Online]

24AM958	HUMAN-CENTERED GEN AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Understand the principles and foundations of Human-Centered AI (HCAI) and its societal impact.
- Analyze the design frameworks and guidelines for developing safe, reliable, and trustworthy AI systems.
- Explore various design metaphors such as intelligent agents, supertools, and human-AI collaboration models.
- Examine governance, ethical, and regulatory aspects involved in deploying AI systems.
- Evaluate emerging trends and future directions in Human-Centered Generative AI applications.

UNIT I	FOUNDATIONS OF HUMAN-CENTERED AI	9

Introduction to Human-Centered AI: Expectations and scope - Philosophical Foundations: Rationalism vs Empiricism in AI - Humans vs Computers: Cognitive and functional differences - Impact of AI on Employment: Automation and workforce concerns.

UNIT II	HUMAN-CENTERED AI FRAMEWORK	9

Introduction to HCAI Framework and levels of automation - Characteristics of Reliable, Safe, and Trustworthy AI systems - Two-Dimensional HCAI Framework: Human control vs automation - Designing Human-in-the-loop systems - Design guidelines for HCAI systems - Case studies and real-world examples

UNIT III	DESIGN METAPHORS IN HUMAN-CENTERED AI	9

Goals of AI research: Scientific and innovation perspectives - Intelligent Agents vs Supertools - AI as Teammates and Tele-bots - Assured autonomy and human control centers - Social robots and active appliances.

UNIT IV	GOVERNANCE AND ETHICAL STRUCTURES	9

Bridging ethics to practice in AI systems - Reliable system design using software engineering practices - Building safety culture through business strategies -
Certification and independent oversight mechanisms - Government regulations and policy frameworks.

UNIT V	FUTURE DIRECTIONS IN HUMAN-CENTERED AI	9

Driving the future of Human-Centered AI - Assessing trustworthiness in AI systems - AI applications for elderly care and assistive technologies - Human-AI collaboration in sensitive domains - Challenges in scaling HCAI systems.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the concepts, foundations, and challenges of Human-Centered AI.
CO2: Analyze AI systems using HCAI frameworks for reliability, safety, and trustworthiness.
CO3: Apply appropriate design principles to develop human-centered AI solutions
CO4: Evaluate different human-AI interaction models and design metaphors in real-world scenarios.
CO5: Assess ethical, legal, and governance issues in AI deployment.
CO6: Propose innovative and responsible AI solutions addressing societal needs using HCAI principles.

TEXT BOOK:

3. Ben Shneiderman. Human-Centered AI, Oxford University Press,2022

REFERENCES:

1. Stuart Russell. Human Compatible: Artificial Intelligence and the Problem of Control, Viking Press, 2019

INTELLIGENT HEALTHCARE

24AM936	AI AND ML FOR HEALTHCARE	L	T	P	C
		2	0	2	3

OBJECTIVES:

- To gain a deep insight into the key concepts of AI and Big data for healthcare.
- To familiarize the principles of drug discovery and molecular modeling.
- To learn the various techniques of machine intelligence for Cancer prediction.
- To explore the recent trends in medical imaging.
- To understand the Remote patient monitoring and AI assisted surgery techniques.

UNIT I	CURRENT HEALTHCARE, BIG DATA, AND MACHINE LEARNING	6+6

Current healthcare practice- Value-based treatments and healthcare services- Increasing data volumes in healthcare – Analytics of healthcare data – The new age of healthcare- Precision medicine- Artificial intelligence and medical visualization- Intelligent personal health records-
Robotics and artificial intelligence-powered devices- Ambient assisted living- Success factors for artificial intelligence in healthcare

List of Lab Exercises:

1. Perform Diagnostic Analytics for a medical data set
2. Perform Prescriptive Analytics for a medical data set

UNIT II	DRUG DISCOVERY AND MOLECULAR MODELING	6+6

Introduction - The scope of artificial intelligence in drug discovery- Types of machine learning in artificial intelligence- Molecular modeling and databases in AI for drug molecules- ML methods in molecular modeling- Drug characterization- Drug design for neuroreceptors using ANN techniques- Use of deep learning in drug design

List of Lab Exercises:

1. Perform drug discovery Analytics using pharmaceutical data set
2. Perform Molecular Modeling Analytics using Molecular Modeling DataBase

UNIT III	CANCER DIAGNOSTICS AND TREATMENT DECISIONS	6+6

Background- AI, ML, and deep learning in cancer- Determine cancer susceptibility- Enhanced cancer diagnosis and staging- Predict cancer treatment response- Predict cancer recurrence and survival- Personalized cancer pharmacotherapy

List of Lab Exercises:

1. Perform Cancer Detection Analytics using a medical data set.
2. Perform Cancer Treatment Decision Analytics using a medical data set.

UNIT IV	ARTIFICIAL INTELLIGENCE FOR MEDICAL IMAGING	6+6

Introduction – AI in radiology/medical imaging – overcoming the hurdles - X-rays and AI in medical imaging - Ultrasound and AI in medical imaging- Application of AI in medical imaging - The development of AI in medical devices - Limitations of AI in medical devices - The future frontiers of AI in medical devices

List of Lab Exercises:

1. Perform Xray Image Analysis using a medical data set.
2. Perform Ultrasound Analysis using a medical data set.

UNIT V	REMOTE PATIENT MONITORING USING AI	6+6

Introduction - Deploying patient monitoring - The role of AI in remote patient monitoring - Diabetes prediction and monitoring using AI - Cardiac monitoring using AI - Neural applications and remote patient monitoring - Artificial intelligence assisted surgery- Preoperative – Intraoperative - Postoperative

List of Lab Exercises:

1. Develop a IOT based Remote Patient Monitoring system Project

TOTAL: 30+30=60 PERIODS

OUTCOMES:

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

CO1: Elaborate the key concepts of AI and Big data for healthcare.
CO2: Illustrate the principles of drug discovery and molecular modeling.
CO3: Implement various techniques of machine intelligence for Healthcare applications.
CO4: Identify the recent trends in medical imaging.
CO5: Understand the Remote patient monitoring system.
CO6: Apply various algorithms of AI and ML to solve Healthcare problems.

TEXT BOOKS:

1. Adam Bohr, Kaveh Memarzadeh, Artificial Intelligence in Healthcare, Academic Press is an imprint of Elsevier, 2020.

REFERENCES:

1. Arjun Panesar ,Machine Learning and AI for Healthcare: Big Data for Improved Health Outcomes, APress, 2019.
2. Rangaraj M. Rangayyan, Biomedical Image Analysis, 2004.
3. Ranjay Krishna, "Computer Vision: Foundations and Applications", Standford University, 2017.
4. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer 2011.
5. S.N. Sivanandam, S.N. Deepa, Principles of Soft Computing, 3rd Edition, Wiley, 2018.

LIST OF EQUIPMENTS:

Systems with Anaconda, Jupyter Notebook, Python

24AM937	MEDICAL IMAGE ANALYSIS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand of various medical imaging modalities.
- Explore advanced deep learning techniques for medical image analysis.
- Develop solutions by preprocessing medical images, implementing machine learning and deep learning algorithms.
- Examine the ethical implications and societal impact of deploying machine learning models in healthcare.
- Elaborate on recent advances and research trends in machine intelligence for medical image analysis.

UNIT I	INTRODUCTION TO MEDICAL IMAGING	9

Overview of medical imaging modalities -MRI, CT, X-ray, Ultrasound-Basics of image acquisition, processing, and visualization in medical imaging-Challenges and importance of medical image analysis-Introduction to common medical imaging datasets.

UNIT II	FUNDAMENTALS OF MACHINE LEARNING	9

Introduction to machine learning concepts-Supervised, unsupervised, and semi-supervised learning-Feature extraction and feature selection techniques-Evaluation metrics for machine learning models.

UNIT III	DEEP LEARNING FUNDAMENTALS	9

Basics of artificial neural networks (ANNs)-Convolutional Neural Networks (CNNs) for image analysis-Recurrent Neural Networks (RNNs) for sequential data analysis-Transfer learning and pre-trained models.

UNIT IV	MEDICAL IMAGE PREPROCESSING	9

Image preprocessing techniques specific to medical images -noise reduction, normalization-Segmentation techniques-thresholding, region growing-Registration and alignment of medical images-Data augmentation for medical image datasets

UNIT V	MEDICAL IMAGE ANALYSIS	9

Classification of medical images using machine learning algorithms-Object detection and localization in medical images-Case studies and applications of machine learning in medical image analysis.

Overview of deep learning architectures for medical image analysis-Semantic segmentation for medical images-Generative models for medical image synthesis-Ethical considerations and challenges in deploying deep learning models in healthcare.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Demonstrate a comprehensive understanding of various medical imaging modalities.

CO2: Apply machine learning and deep learning techniques.

CO3: Develop solutions by preprocessing medical images, implementing machine learning and deep learning algorithms.

CO4: Understand the ethical considerations and regulatory requirements associated with deploying machine intelligence models in healthcare settings.

CO5: Elaborate on recent advances and research trends in machine intelligence for medical image analysis.

CO6: Illustrate the applications of ML and DL in medical image analysis.

TEXT BOOKS:

1. Le Lu, Yefeng Zheng, Gustavo Carneiro, Lin Yang , Deep Learning and Convolutional Neural Networks for Medical Image Computing Precision Medicine, High Performance and Large-Scale Dataset, Springer, 2017.
2. Atam P. Dhawan, “Medical Image Analysis", Wiley Publications, 2010.

REFERENCES:

1. Ton J. Cleophas and Aeilko H. Zwinderman , Machine Learning in Medicine - A Complete Overview", Springer, 2015.

2. Nadine Barrie Smith and Andrew Webb, “Introduction to Medical Imaging: Physics, Engineering and Clinical Applications", Cambridge University Press, 2010.

24AM938	CLINICAL DATA SCIENCE	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Discuss standards to generate clinical data from electronic medical records.
- Elaborate various Modelling methods on Clinical Data.
- Illustrate methods to perform clinical data analysis using various data analysis techniques.
- Interpret clinical data analysis to support decision making.
- Apply statistics to improve the quality of decision making.
- Develop applications using Clinical Data.

UNIT I	INTRODUCTION	9

Data Sources – Electronic Medical Records – Laboratory Information Management Systems - GDPR – Data Types – Data Standards – Big Clinical Data – Data Landscape – Standardizing Clinical Data.

UNIT II	CLINICAL DATA TO MODELS	9

Preparing Data for Predictive Modelling – Designs for Model Development – Sample size – Missing Data – Time-Domain Processing – Frequency-Domain Processing – Prediction Modelling Methodology.

UNIT III	CLINICAL DATA ANALYSIS	9

Clinical Trials – Classifications – Discrete Data Analysis – Failure-time Data Analysis – Quantitative Data Analysis – Multiplicity Analysis.

UNIT IV	MEDICAL STATISTICS	9

Prove Prior Hypothesis – Improve the quality of research – Testing Randomness – Quality criteria.

UNIT V	APPLICATIONS	9

Clinical Decision Support System – Types – Challenges - Best Knowledge & Continuous Improvement of Knowledge and CDSS Methods – Mobile CDSS – Care Process – Operational Excellence – Process Mining - Sociotechnical Systems & Leadership - Value-Based Health Care Supported by Data Science.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Generate clinical data from electronic medical records.
CO2: Employ various Modelling methods on Clinical Data.
CO3: Perform clinical data analysis using various data analysis techniques.
CO4: Interpret clinical data to support decision making.
CO5: Apply statistics to improve the quality of decision making.
CO6: Develop applications using Clinical Data.

TEXT BOOKS:

1. Pieter Kubben, Michel Dumontier, Andre Dekker, Fundamentals of Clinical Data Science, Springer, 2019.
2. Ton J. Cleophas, Aeilko H. Zwinderman, Understanding Clinical Data Analysis: Learning Statistical Principles from Published Clinical Research, Springer, 2016.

REFERENCES:

1. Aeilko H. Zwinderman, Ton J. Cleophas, Machine Learning in Medicine - A Complete Overview, Springer, 2021.

24AM939	DEEP LEARNING IN GENOMICS AND LIFE SCIENCES	L	T	P	C
		3	0	0	3

OBJECTIVES:

- Represent molecules and proteins as features for building machine learning models.
- Emphasize how to extract interpretable, biological insights from deep learning models.
- Illustrate the applications of deep learning in genomics.
- Analyze different models for Genomic applications.
- Employ various deep learning tools for genomics.
- Apply GANs for improving the models.

UNIT I	MACHINE LEARNING IN GENOMICS	9

Machine Learning for Genomics - Biopython – Genomics Data Analysis – Genome – Genome sequencing – Sanger sequencing of nucleic acids – Evolution of next generation sequencing – Analysis – steps – Calculating GC content – nucleotide content- Dinucleotide content – Modelling – Motif finder.

Case Study: Sequence Analysis of Covid-19

UNIT II	BIOPHYSICAL MACHINE LEARNING	9

Molecule - Molecular Bonds - Molecular Graphs - Molecular Conformations - Chirality of Molecules - Featurizing a Molecule - Graph Convolutions - Protein Structures - Protein Sequences - Biophysical Featurizations - Grid Featurization - Atomic Featurization.

Case Study: Analyzing the PDBBind Dataset.

UNIT III	DEEP LEARNING FOR GENOMIC APPLICATIONS	9

DNNs for Genomics – workflow for Genomics – Protein structure predictions – Regulatory genomics – Gene regulatory Networks – Single-cell RNA sequencing – Deep learning libraries for genomics.

Case Study: Disease prediction

UNIT IV	CNN AND RNN FOR GENOMICS	9

Transfer Learning – CNNs for Genomics – Applications – Deep Bind – DeepInsight – DeepChrome – DeepVariant – Applications and use cases of RNNs in Genomics – DeepNano – ProLanGo – DanQ – Autoencoders for genomics – Gene Expression.

Case Study: Predicting Gene expression from TCGA pan-cancer RNA-S using denoising autoencoders.

UNIT V	MODEL IMPROVEMENT	9

GANs for Improving Models – Difference between Discriminative and Generative Models – Challenges – synthetic data – Applications – Analysis of ScRNA-Seq data – Generation of DNA.

Case Study: Personalized Medicine

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Represent molecules and proteins as features for building machine learning models.
CO2: Extract interpretable, biological insights from deep learning models.
CO3: Illustrate the applications of deep learning in genomics.
CO4: Analyze different models for Genomic applications.
CO5: Employ various deep learning tools for genomics.
CO6: Apply GANs for improving the models.

TEXT BOOKS:

1. Upendra Kumar Devisetty, Deep Learning for Genomics: Data-driven approaches for genomics applications in life sciences and biotechnology, packt Publications, 2022.
2. Bharath Ramsundar, Peter Eastman, Patrick Walters, Vijay Pande, Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery & More, O'Reilly, 2019.

REFERENCES:

1. Sanjiban Sekhar Roy, Y.-H. Taguchi, Handbook of Machine Learning Applications for Genomics, Springer, 2022.

2. Shailza Singh, Machine Learning and Systems Biology in Genomics and Health, Springer, 2022.

24AM940	SMART AND INTERACTIVE HEALTHCARE TECHNOLOGIES	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Illustrate the need and challenges of personalized healthcare.
- Explore the basic aspects of telehealth and telemedicine.
- Enumerate mHealth evolution, regulation and applications.
- Demonstrate the use of virtual reality and games in healthcare.
- Elaborate the importance of IoT in healthcare through its applications.

UNIT I	PERSONALISED HEALTHCARE	9

Personalization of healthcare: the relationship between data, Digital technologies and advanced analytics – Digital health measures – Examples in digital health technologies in clinical research - Examples in digital health technologies in care delivery pathway –Challenges in bringing digital health technologies to market –Challenges in adoption of digital health technologies.

UNIT II	TELEHEALTH AND TELEMEDICINE	9

Telemedicine versus telehealth – Definitions - Technology vs services – Telemedicine technological requirements – Telehealth technological requirements – Distant health examples – Smart medical shirts – Haptic platform – Overgrown cities – Rural health – Satellite telehealth – Telemedicine critical technologies – Present challenges and benefits – Groundwork for a good telehealth application – Enabling telehealth for existing medical application – Case study – Panic disorder – Case study – Diabetes telehealth framework – Case study – telehealth support for unit care – Medicolegal, ethical and regulatory guidelines pertaining to telehealth.

UNIT III	M-HEALTH	9

Evolution from telemedicine to m-Health – Initial and recent applications – Mobile apps for mHealth - Overview of mHealth apps – Regulation of mHealth apps - Cloud computing definition and selected applications – closed loop solutions for personalized health interventions – Challenges in sensor design and fabrication – Challenges in mining and managing Big health data – Common mHealth and ICT applications – Evidence for mHealth impact –New frontiers in mHealth - Case study –Sleep problems and their implications.

UNIT IV	VIRTUAL REALITY AND GAMES FOR HEALTHCARE	9

Augmenting mental healthcare – Mobilizing services with virtual reality – Pain – Anxiety and phobias – Stress management – Rehabilitation – Games for improving healthcare – Homo Ludens – Learning through challenges and fun – Physical and functional fidelity – Games for health – Rehabilitation – Crowdsourcing science – Gaming doctor – Games in official Medical programs – Games in skills training outing operating room – Financial and Ethical aspects.

UNIT V	IOT FOR HEALTHCARE	9

Concept of IoT-Based Healthcare Technology – Ambient intelligence in Healthcare Technologies – Benefits – Challenges – Data handling and resource management – Security and Privacy – Interoperability – Stake holder collaboration and implementation – Technologies that enable IoT – Healthcare – Internet of Medical things – Applications of IoT in Healthcare – Benefits – Challenges.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Illustrate the need and challenges of personalized healthcare.
CO2: Apply basic aspects of telehealth and telemedicine.
CO3: Demonstrate M-Health evolution, regulation and applications.
CO4: Elaborate the use of virtual reality and games in healthcare.
CO5: Elaborate the importance of IoT in healthcare through its applications.
CO6: Apply smart and interactive technologies for healthcare applications.

TEXT BOOKS:

1. Halit Eren and John G Webster, “Telemedicine and Electronic Medicine”, CRC Press, Taylor and Francis Group, New York, 2nd edition, 2016.
2. Shabbir Syed-Abdul, Xinxin Zhu, Luis Fernandez-Luque, “Digital Health: Mobile and Wearable

Devices for participatory Health Applications", Elsevier, Cambridge, 2021.
3. Shashi Gogia, "Fundamentals of Telemedicine and Telehealth", Elsevier, Cambridge, USA, 1st Edition, 2020.

REFERENCES:

1. Homero Rivas and Katarzyna Wac, "Digital Health: Scaling Healthcare to the World", Health Informatics, Springer, Switzerland, 2018.
2. Nishu Gupta and Sara Paiva, "IoT and ICT for Healthcare Applications", Springer Innovations in Communication and Computing, 2020.

24AM941	BIO-INFORMATICS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand and develop models for Biological Data.
- Implement image processing Techniques to Bioinformatics Data
- Implement Micro Array analysis over Genome Expression.
- Understand the study of simbiology.
- Understand the pharmacokinetic modeling.
- Understand the working model of biological data in Matlab.

UNIT I	INTRODUCTION	9

Overview of Bioinformatics Technologies – Structural Bioinformatics – Data Format and Processing – Secondary Resources and Applications – Role of Structural Bioinformatics – Biological Data Integration System

UNIT II	BIOINFORMATICS TOOL BOX	9

Sequence Analysis – NGS – Graph Theory – Gene Ontology – Importing Data and Deploying.

UNIT III	BIOLOGICAL DATA ANALYSIS	9

Microarray Data Analysis – Mass Spectrometry Data Analysis – Statistical Classification of Biological Data.

UNIT IV	IMAGE PROCESSING	9

Key Features of Image Processing – Importing and Exporting Images – Image File Formats and Format Conversion – Pre and Post Processing Images – Spatial Transformations and Image Registration – Microarray Image Analysis.

UNIT V	SYSTEMS BIOLOGY	9

Basics of Enzyme Kinetics – Kinetic Laws – Modeling Biological System: Simulation, Sensitivity Analysis, Parameter Estimation using Simbiology – Pharmacokinetic Modeling: Simulation, Population Study – Model of the Yeast Heterotrimeric G Protein Cycle and Glycoly .

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Develop models for Biological Data.
CO2: Implement image processing Techniques to Bioinformatics Data
CO3: Implement Micro Array analysis over Genome Expression.
CO4: Understand the study of simbiology.
CO5: Illustrate the pharmacokinetic modeling.
CO6: Elaborate the working model of biological data in Matlab.

TEXT BOOKS:

1. Yi-Ping Phoebe Chen(Ed),"Bioinformatics Technologies", Springer Publications, 2015
2. G. Alterovitz, M. F. Ramoni, "Systems Bioinformatics: An Engineering Case-Based Approach", Artech House, 2017.

REFERENCES:

1. Michael R. King, Nipa A. Mody, "Numerical and Statistical Methods for Bioengineering: Applications in MATLAB", Cambridge University Press, 2011.
2. John L. Semmlow, "Bio signal and Medical Image Processing", CRC Press, 2004.
3. Frank C. Hoppensteadt, Charles S. Peskin, "Modeling and Simulation in Medicine and Life Sciences", Springer, 2010.
4. C. Gibas, Per Jambeck, "Developing bio- informatics computer skills", O'Reilly Media, 2001

24AM942	COMPUTATIONAL NEUROSCIENCE	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand what nervous systems do and determine how they function.
- To explore the computational principles governing various aspects of vision, sensory-motor control, learning, and memory.
- To analyze neural models.
- To learn to extract information through neural encoding and decoding.
- To investigate models of synaptic plasticity and learning in the brain.

UNIT I	NEURAL ENCODING	9

Firing Rates and Spike Statistics: Introduction- Spike Trains and Firing Rates - What Makes a Neuron Fire? Spike-Train Statistics – The Neural Code

Reverse Correlation and Visual Receptive Fields – Estimating Firing Rates Introduction to the Early Visual System Reverse-Correlation Methods: Simple Cells Static Non linearities: Complex Cells - Receptive Fields in the Retina and LGN Constructing Visual Receptive Fields

UNIT II	NEURAL DECODING AND INFORMATION THEORY	9

Discrimination - Population Decoding - Spike-Train Decoding

Information Theory: Entropy and Mutual Information – Information and Entropy Maximization – Entropy and Information for Spike Trains

UNIT III	MODEL NEURONS	9

Phase Plane Analysis – I - Phase Plane Analysis – II - Analyzing HHE – Bifurcations - Other Point Models – Levels of Neuron Modeling-Conductance-Based Models – The Cable Equation- Multi-compartment models

UNIT IV	NETWORK MODELS	9

Firing Rate Models – Feedforward Networks – Recurrent Networks – Excitatory-Inhibitory Networks – Stochastic Networks

UNIT V	PLASTICITY	9

Synaptic Transmission and Synaptic Strength - Ways of Modification of Synaptic Strength - Types of Plasticity - Short Term Plasticity - Long Term Plasticity – Computational Implications

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the fundamentals of neural encoding.
CO2: Apply neural encoding techniques.
CO3: Use Information Theory to decode neural signals.
CO4: Analyze and model the dynamics of neurons.
CO5: Design and analyze neural networks.
CO6: Implement the concepts of synaptic plasticity.

TEXT BOOKS:

1. Dayan, Peter, and L. F. Abbott, Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems. Cambridge, MA: MIT Press, 2005. ISBN: 9780262041997.
2. Paul Miller, An Introductory Course in Computational Neuroscience, MIT Press, 2018.

REFERENCES:

1. Signal and Systems, Alan V. Oppenheim, Alan S. Willsky, Syed Hamid Nawab Prentice Hall, 1997.
2. Methods in Neuronal Modeling, Second Edition From Ions to Networks , Edited by Christof Koch and Idan Segev, MIT Press
3. Ionic Channels of Excitable Membranes, Second Edition, Bertil Hille, Sinauer Associates Inc.,1992
4. NPTEL: Computational Neuroscience - Course (nptel.ac.in)

24AM943	COMPUTATIONAL BIOLOGY AND SEQUENCE ANALYSIS	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- To understand computational approaches for biological data analysis
- To learn string algorithms for genome sequence processing
- To apply sequence alignment techniques in bioinformatics
- To analyze genome assembly and evolutionary relationships
- To explore probabilistic and machine learning models in genomics

UNIT I	INTRODUCTION TO COMPUTATIONAL BIOLOGY	9

Overview of molecular biology and genetics – Biological sequences (DNA, RNA, proteins) – Genomic databases and tools – Challenges in large-scale biological data – Applications in bioinformatics and healthcare.

UNIT II	STRING ALGORITHMS FOR SEQUENCE ANALYSIS	9

Exact string matching – Z-algorithm – Suffix arrays and construction – Suffix trees and construction – Burrows-Wheeler Transform (BWT) – Applications in sequence searching.

UNIT III	SEQUENCE ALIGNMENT TECHNIQUES	9

Pairwise sequence alignment – Dynamic programming – Edit distance – Global alignment – Local alignment – Gap penalties – Statistical significance – Heuristic-based alignment methods.

UNIT IV	GENOME RECONSTRUCTION AND GRAPH ALGORITHMS	9

Genome assembly problem – Shortest common superstring – Greedy methods – de Bruijn graphs – Overlap graphs – Applications in mutation detection and disease diagnosis.

UNIT V	ADVANCED TOPICS IN COMPUTATIONAL GENOMICS	9

Multiple sequence alignment – Evolutionary tree construction – Hidden Markov Models (HMM) – Machine learning models for biological sequences – Pangenome graph

TOTAL: 45 Periods

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

CO1: Explain biological data and computational biology concepts
CO2: Apply string algorithms for sequence analysis
CO3: Implement sequence alignment techniques
CO4: Analyze genome reconstruction using graph algorithms
CO5: Apply probabilistic and ML models in genomics.

TEXT BOOKS:

1. Dan Gusfield, Algorithms on Strings, Trees, and Sequences, Cambridge University Press.
2. Veli Mäkinen et al., Genome-Scale Algorithm Design, Cambridge University Press.

REFERENCES:

1. Richard Durbin et al., Biological Sequence Analysis, Cambridge University Press
2. Srinivas Aluru, Handbook of Computational Molecular Biology, Chapman & Hall
3. Thomas H. Cormen et al., Introduction to Algorithms, MIT Press

24AM944	APPLIED ML FOR LIFE SCIENCES	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- To understand the fundamentals of machine learning in life sciences
- To learn preprocessing and feature engineering for biological data
- To apply supervised and unsupervised learning techniques
- To develop models for genomics, healthcare, and drug discovery
- To understand deep learning applications in biomedical fields

UNIT I	INTRODUCTION TO MACHINE LEARNING IN LIFE SCIENCES	9

Overview of Machine Learning – Types of Machine Learning – ML workflow – Biological data types: genomics, proteomics, clinical data – Data sources and challenges – Applications in healthcare, bioinformatics, and medical diagnosis.

UNIT II	DATA PREPROCESSING AND FEATURE ENGINEERING	9

Data cleaning and normalization – Handling missing and noisy data – Feature extraction from biological datasets – Dimensionality reduction – Principal Component Analysis (PCA) – Data visualization techniques.

UNIT III	SUPERVISED LEARNING METHODS	9

Linear regression – Logistic regression – Decision trees – Random forests – Support Vector Machines – Model training and validation – Performance metrics: accuracy, precision, recall, F1-score – Applications in disease prediction.

UNIT IV	UNSUPERVISED LEARNING METHODS	9

Clustering techniques: K-means, hierarchical clustering – Pattern discovery in biological data – Gene expression analysis – Anomaly detection – Association rule mining.

UNIT V	DEEP LEARNING AND APPLICATIONS IN LIFE SCIENCES	9

Neural networks – Deep learning basics – Convolutional Neural Networks (CNN) for medical imaging – Sequence models for DNA/RNA analysis – Drug discovery using ML – Precision medicine – Ethical issues in healthcare AI.

TOTAL: 45 Periods

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

CO1: Explain machine learning concepts, workflow, and types of biological data used in life sciences applications.
CO2: Apply data preprocessing and feature engineering techniques such as normalization, dimensionality reduction, and visualization on biological datasets.
CO3: Implement supervised learning algorithms to solve classification and regression problems in healthcare and bioinformatics.
CO4: Analyze biological data using unsupervised learning techniques such as clustering, association rule mining, and anomaly detection.
CO5: Apply deep learning techniques to develop models for biomedical applications such as medical imaging, genomics, and drug discovery.

TEXT BOOKS:

1.Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

2.Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012

REFERENCES:

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer
2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn & TensorFlow*, O'Reilly
3. Lesk, A.M., *Introduction to Bioinformatics*, Oxford

QUANTUM INTELLIGENCE AND COMPUTATIONAL SCIENCES (QI&CS)

24AM947	QUANTUM MECHANICS AND QUANTUM CIRCUITS FOR COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Introduce foundational quantum mechanics concepts for quantum computing.
- Build proficiency in quantum mathematical tools and representations.
- Teach design and analysis of quantum circuits with entanglement.
- Explain the quantum circuit model and its limitations.
- Compare classical and quantum computational models and advantages.

UNIT I	BASICS OF QUANTUM MECHANICS-I	9

Basic elements of QM: Preparation and Measurement (introduction)-Quantum state: mathematical description of preparation procedure-Projector: mathematical description of measurement procedure.

UNIT II	BASICS OF QUANTUM MECHANICS-II	9

Basic elements of QM: Preparation and Measurement-Mixed state -Distinct measurements - Quantum evolution and superposition-Quantum entanglement Quantum channel: Completely Positive Trace Preserving (CPTP) map.

UNIT III	QUANTUM CIRCUITS-I	9

The basics of the quantum circuit model -Working with one qubit and the Bloch sphere-Qubit-Dirac notation and inner products -One-qubit quantum gates-The Bloch sphere and rotations.

UNIT IV	QUANTUM CIRCUITS-II	9

Working with two qubits and entanglement -Two-qubit states-Two-qubit gates: tensor-The CNOT gate- Entanglement-The no-cloning theorem- Controlled gates

UNIT V	AN OVERVIEW OF COMPUTATION	9

Classical Information and Computation-Models of Computation-FSMs and Turing Machine-Computational Problems-Algorithms and Optimization Problems-Computational Complexity-Complexity Computational Model

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain fundamental quantum mechanics concepts like state preparation, measurement, superposition, entanglement, and quantum channels.

CO2: Represent qubits using Dirac notation and Bloch sphere and apply single-qubit gates.

CO3: Construct and analyze multi-qubit quantum circuits including CNOT and controlled gates.

CO4: Demonstrate understanding of entanglement creation and limitations like the no-cloning theorem.

CO5: Compare classical computational models (FSM, Turing machines) with quantum circuit model and complexity.

TEXT BOOKS:

1. Michael A. Nielsen and Isaac L. Chuang, "Quantum Computation and Quantum Information", 10th Anniversary edition, Cambridge University Press, 2010.
2. Thomas G. Wong, "Introduction to Classical and Quantum Computing", 1st edition, Rooted Grove Books, 2022.

REFERENCES:

1. David J. Griffiths and Darrell F. Schroeter, "Introduction to Quantum Mechanics", 3rd edition, Cambridge University Press, 2018.
2. Chris Bernhardt, "Quantum Computing for Everyone", 1st edition, MIT Press, 2019.
3. R. Shankar, "Principles of Quantum Mechanics", 2nd edition, Springer, 1994.
4. Eleanor G. Rieffel and Wolfgang H. Polak, "Quantum Computing: A Gentle

Introduction", 1st edition, MIT Press, 2011.
5. Ciaran Hughes et al., "Quantum Computing for the Quantum Curious", 1st edition, Springer, 2021.

24AM948	QUANTUM OPTIMIZATION AND QUANTUM MACHINE LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand the principles of quantum-inspired computing, variational quantum algorithms, and their applications in optimization and machine learning.
- Learn key algorithms such as VQE, QAOA, and HHL for solving real-world problems like electronic structure, combinatorial optimization, and linear systems.
- Explore hybrid quantum-classical approaches and challenges in near-term quantum devices (e.g., barren plateaus).
- Gain insight into quantum models of machine learning, including quantum neural networks and their potential advantages.

UNIT I	Quantum Inspired Computing and VQA	9

Introduction-Ising Model-Quantum Annealing-Simulated Annealing-Tensor Networks-Variational Quantum Algorithms-Building blocks of a variational quantum algorithm (1)-Building blocks of a variational quantum algorithm (2) Setup: Variational quantum algorithm (VQA)-VQA Example: VQE and QAOA Challenge: Barren Plateaus-Current-generation application landscape for VQAs.

UNIT II	Variational Quantum Eigensolver	9

Introduction-Hybrid Quantum Classical Algorithms-Variational Quantum Algorithms-Variational Quantum Eigensolver-Electronic Structure Problem Barren Plateus

UNIT III	Quantum Approximate Optimization Algorithm	9

Introduction-The Combinatorial Optimization Problem (COP) ≡ QUBO Quantum formulation of the COP-Quantum Approximate Optimization Algorithm (QAOA) for the COP-Single (p = 1) layer QAOA-Single (p = 1) layer QAOA for n = 3 qubits-Computing gradient on a quantum computer by the parameter sift rule-Traveling Salesman Problem (TSP)-Quantum (QAOA) formulation of the TSP.

UNIT IV	Quantum Machine Learning	9

Introduction-Why QML-Quantum models of ML algorithms-Variational quantum machine learning-Building neural networks in quantum model-A single qubit neural network-A single-qubit neural network for classification problem Hybrid quantum-classical neural network for molecular energy problem.

UNIT V	HHL quantum algorithm for solving a system of linear equations	9

Introduction-A system of linear equations-The HHL quantum algorithm - Data Loading : State Preparation-The HHL quantum algorithm: an example

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain and implement building blocks of variational quantum algorithms (VQA, VQE, QAOA).

CO2: Formulate combinatorial optimization problems (e.g., TSP) in quantum terms and solve them using QAOA.

CO3: Apply variational quantum eigensolver (VQE) to problems like molecular energy estimation and understand issues like barren plateaus.

CO4: Describe quantum machine learning concepts and build simple hybrid quantum-classical neural networks.

CO5: Understand and outline the HHL algorithm for solving linear equations on quantum computers.

TEXT BOOKS:

1. Elías F. Combarro and Samuel González-Castillo, “A Practical Guide to Quantum Machine Learning and Quantum Optimization: Hands-on Approach to Modern Quantum Algorithms”, Packt Publishing, 2023.

2. Maria Schuld and Francesco Petruccione, “Machine Learning with Quantum Computers”, 2nd edition, Springer, 2021.

REFERENCES:

1. Michael A. Nielsen and Isaac L. Chuang, “Quantum Computation and Quantum Information”, 10th Anniversary edition, Cambridge University Press, 2010.
2. M. Cerezo et al., “Variational quantum algorithms”, Nature Reviews Physics, 2021.
3. Peter Wittek, “Quantum Machine Learning: What Quantum Computing Means to Data Mining”, Academic Press, 2014.
4. Davide Pastorello, “Concise Guide to Quantum Machine Learning”, Springer, 2023.
5. Osvaldo Simeone, “An I ntroduction to Quantum Machine Learning for Engineers”, Foundations and Trends in Signal Processing, 2022.

24AM949	QUANTUM ALGORITHMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- Understand foundational quantum algorithms and their speedups.
- Build and analyze quantum circuits for key algorithms.
- Learn mathematical tools like quantum Fourier transform.
- Explore impacts on cryptography and search problems.
- Compare quantum and classical computational complexities.

UNIT I	Quantum Algorithms- Deutsch Algorithm	9

Creating a framework to build quantum algorithms-Setting up Deutsch Problem Functions as Oracles-Quantum Circuit U to implement function f(x)-Classical solution for solving the Deutsch Problem-Quantum query complexity- Quantum phenomenon as a resource in algorithm design- Building a quantum circuit to solve Deutsch Problem

UNIT II	Quantum Algorithms- Deutsch-Jozsa Algorithm & Bernstein Vazirani Algorithm	9

Deutsch-Jozsa Problem-Deutsch-Jozsa Algorithm-Bernstein-Vazirani Problem- Bernstein-Vazirani Algorithm

UNIT III	Quantum Algorithms- RSA encryption, Shor factoring algorithm, and quantum Fourier transform-I	9

Introduction-Modular (remainder) arithmetic and multiplicative group of co primes-RSA (Rivest–Shamir–Adleman) encryption-RSA (Rivest–Shamir Adleman) encryption: Example.

UNIT IV	Quantum Algorithms- RSA encryption, Shor factoring algorithm, and quantum Fourier transform-II	9

Prime factorization-Prime factorization is equivalent to order finding-Quantum algorithm for order-finding- Discrete Fourier transform (DFT)-Fast Fourier transform (FFT)-Quantum Fourier transform (QFT)-Unknown phase (eigenvalue) estimation-Amplitude estimation: Grover + Shor algorithm

UNIT V	Quantum Algorithms- Quantum Database Search: Grover's Algorithm	9

Introduction- Grover's algorithm : Quantum circuit-1- Grover's algorithm : Quantum circuit-2-Visualization and Applications- Bell State-Single Qubit Gates-Multi Qubit Gates.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO 1: Explain the principles of key quantum algorithms including Deutsch, Deutsch-Jozsa, Bernstein-Vazirani, Shor's, and Grover's.

CO 2: Design and analyze quantum circuits for oracles, quantum Fourier transform, phase estimation, and amplitude amplification.

CO 3: Compare quantum and classical query complexities for specific problems and identify quantum resources enabling speedups.

CO 4: Apply quantum algorithms to real-world scenarios such as RSA encryption breaking and unstructured database search.

CO 5: Implement basic quantum algorithms conceptually and recognize their potential impacts on cryptography and optimization

TEXT BOOKS:

1. Eleanor G. Rieffel and Wolfgang H. Polak, "Quantum Computing: A Gentle Introduction", MIT Press, 2011.
2. Michael A. Nielsen and Isaac L. Chuang, "Quantum Computation and Quantum Information", Cambridge University Press, 2010.

REFERENCES:

3. Elías F. Combarro and Samuel González-Castillo, "A Practical Guide to Quantum Machine Learning and

	Quantum Optimization: Hands-on Approach to Modern Quantum Algorithms", Packt Publishing, 2023.
4.	Richard J. Lipton and Kenneth W. Regan, "Introduction to Quantum Algorithms via Linear Algebra", Second Edition, MIT Press, 2021.
5.	Noson S. Yanofsky and Mirco A. Mannucci, "Quantum Computing for Computer Scientists", Cambridge University Press, 2008.
6.	Chris Bernhardt, "Quantum Computing for Everyone", MIT Press, 2019.
7.	N. David Mermin, "Quantum Computer Science: An Introduction", Cambridge University Press, 2007.

OPEN ELECTIVE (Offered to Other Departments by ADS)

24AM001	FOUNDATIONS OF DEEP LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To outline the basics of deep neural networks.
- To discuss advanced deep learning models.
- To discuss CNN and RNN architectures of deep neural networks.
- To elaborate autoencoders in neural networks.
- To discuss the deep generative models.

UNIT I	DEEP NETWORKS	9

Challenges motivating deep learning - Deep feedforward networks - Learning XOR - Gradient based learning - Hidden Units – Architecture Design – Back Propagation – Regularization – Parameter Norm Penalties – Constrained Optimization – Under-Constrained Problems – Dataset Augmentation – Noise Robustness – Semi-Supervised Learning – Multi-Task Learning – Early Stopping – Parameter Tying and Sharing – Bagging and Other Ensemble methods – Dropout – Adversarial Training.

UNIT II	OPTIMIZATION FOR TRAINING DEEP MODELS	9

Pure optimization – Challenges – Basic Algorithms – Parameter initialization Strategies – Algorithms with Adaptive Learning Rates – Approximate Second-Order methods – Optimization Strategies and Meta Algorithms.

UNIT III	CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS	9

Convolution Operation – motivation – Pooling – Infinitely Strong prior – Variants – Structured Output – Data Types – Efficient Convolutional Algorithms – Random or Unsupervised features – Neuroscientific Basis - Deep Learning – Sequence Modelling - Computational Graphs - RNN - Bidirectional RNN – Encoder-Decoder - Sequence to Sequence RNN - Deep Recurrent Networks - Recursive Neural Networks -- Long Term Dependencies; Leaky Units – Strategies for multiple time scales – LSTM and Gated RNNs – Optimization for Long Term Dependencies.

UNIT IV	AUTOENCODERS	9

Autoencoders: Undercomplete autoencoders - Regularized autoencoders – Power, Layer Size and Depth - Stochastic encoders and decoders – Denoising Autoencoders - Learning with autoencoders – contractive Autoencoders – Applications of autoencoders.

UNIT V	DEEP GENERATIVE MODELS	9

Boltzmann Machine – Restricted Boltzmann Machine – Deep Belief Networks – Deep Boltzmann Machines - Boltzmann Machines for Real-Valued Data – Convolutional Boltzmann Machines - Boltzmann Machine for Structured or Sequential Outputs – Directed Generative Nets – Evaluating Generative Models.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of this course, the students will be able to: CO1:
Outline the basics of deep neural networks**.**
CO2: Develop advanced deep learning models.
CO3: Implement CNN and RNN architectures of deep neural networks.
CO4: Interpret autoencoders in neural networks.
CO5: Apply deep generative models to solve real world problems.
CO6: Build deep learning models and evaluate them.

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, ``Deep Learning", MIT Press, 2016.

REFERENCES:

1. Charu C. Aggarwal, ``Neural Networks and Deep Learning: A Textbook", Springer International Publishing, 2018.

2. Yoav Goldberg, ``Neural Network Methods for Natural Language Processing", Synthesis

Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
3. Francois Chollet, ``Deep Learning with Python'', Manning Publications Co, 2018.
4. Josh Patterson, Adam Gibson, ``Deep Learning: A Practitioner's Approach'', O'Reilly Media, 2017.
5. Navin Kumar Manaswi, “Deep Learning with Applications Using Python”, Apress, 2018.

NPTEL:

6. Deep Learning - https://onlinecourses.nptel.ac.in/noc24_ee04/preview

Deep Learning - IIT Ropar - https://onlinecourses.nptel.ac.in/noc24_cs59/preview

24AM002	SOFT COMPUTING	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To learn the basic concepts of Soft Computing.
- To understand artificial neural networks.
- To apply fuzzy systems to solve problems.
- To solve problems using Genetic Algorithms.
- To discuss the various Hybrid algorithms and various Swarm Intelligence algorithms.

UNIT I	INTRODUCTION	9

Neural Networks - Application Scope of Neural Networks - Fuzzy Logic - Genetic Algorithm - Hybrid Systems - Soft Computing - Artificial Neural Network - Evolution of Neural Networks - Basic Models of ANN – Weights – Bias – Threshold – Learning Rate – Momentum Factor – Vigilance Parameter- McCulloch–Pitts Neuron - Linear Separability - Hebb Network.

UNIT II	ARTIFICIAL NEURAL NETWORKS	9

Perceptron Networks - Adaptive Linear Neuron - Multiple Adaptive Linear Neurons - Back-Propagation Network - Radial Basis Function Network - Pattern Association – Auto associative and Hetero associative Memory Networks - Bidirectional Associative Memory (BAM) - Hopfield Networks - Fixed Weight Competitive Nets - Kohonen Self-Organizing Feature Maps.

UNIT III	FUZZY SYSTEMS	9

Fuzzy Logic - Classical Sets (Crisp Sets) - Fuzzy Sets – Fuzzy Relation - Features of the Membership Functions - Fuzzification - Methods of Membership Value Assignments - Defuzzification - Lambda-Cuts for Fuzzy Sets (Alpha-Cuts) - Lambda-Cuts for Fuzzy Relations - Defuzzification Methods – Fuzzy Reasoning – Fuzzy Inference Systems

UNIT IV	GENETIC ALGORITHMS	9

Biological Background - Traditional Optimization and Search Techniques- Genetic Algorithm and Search Space- - Simple GA - General Genetic Algorithm - Operators - Stopping Condition - Constraints - Problem Solving - The Schema Theorem- Classification - Holland Classifier Systems- Genetic Programming - Advantages and Limitations- Applications.

UNIT V	HYBRID SOFT COMPUTING AND SWARM INTELLIGENCE ALGORITHMS	9

Neuro-Fuzzy Hybrid Systems - Genetic Neuro-Hybrid Systems - Genetic Fuzzy Hybrid and Fuzzy Genetic Hybrid Systems - Simplified Fuzzy ARTMAP – Swarm Intelligence Algorithms - Ant Colony Optimization – Artificial Bee Colony – Particle Swarm Optimization – Firefly Algorithm.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the basic concepts of Soft Computing.
CO2: Discuss Artificial neural networks and its applications.
CO3: Apply Fuzzy logic to solve different applications.
CO4: Solving problems using Genetic algorithms.
CO5: Discuss various algorithms in Soft computing with its applications and limitations.
CO6: Use various algorithms in Soft computing to solve real-world problems.

TEXT BOOKS:

3. S. N. Sivanandam , S. N. Deepa, "Principles of Soft Computing", Wiley India Pvt. Ltd., 2nd Edition, 2019.
4. Adam Slovik, “Swarm Intelligence Algorithms: Modification and Applications”, Taylor & Francis, First Edition, 2020.

REFERENCES:

6. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.
7. Kwang H. Lee, First course on Fuzzy Theory and Applications‖, Springer, 2005.
8. N.P. Padhy, S. P. Simon, "Soft Computing with MATLAB Programming", Oxford University Press, 2015.

9. S. Rajasekaran, G. A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications ", PHI Learning Pvt. Ltd., 2017.
10. NPTEL Courses:
    a. Introduction To Soft Computing - https://onlinecourses.nptel.ac.in/noc23_cs40/preview

24AM003	PROMPT ENGINEERING	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- Understand the architecture and inner workings of large language models including tokenization, auto-regressive generation, and the transformer architecture.
- Learn how reinforcement learning from human feedback (RLHF) shapes LLM behavior and the differences between instruct and chat models.
- Develop skills in designing LLM applications by sourcing context, assembling prompts, and managing token limits effectively.
- Explore techniques for taming model outputs using temperature, logprobs, and steering strategies.
- Build conversational agents with tool integration and design multi-step LLM workflows and pipelines.
- Evaluate LLM applications using offline and online testing frameworks, and understand emerging trends in multimodality and ethics.

UNIT I	INTRODUCTION TO PROMPT ENGINEERING AND UNDERSTANDING LLMs	9

Introduction to Prompt Engineering: LLMs Are Magic – Language Models: How Did We Get Here? – Early Language Models – GPT Enters the Scene – Prompt Engineering.
Understanding LLMs: What Are LLMs? – Completing a Document – Human Thought Versus LLM Processing – Hallucinations – How LLMs See the World – LLMs Use Deterministic Tokenizers – LLMs Can't Slow Down and Examine Letters – LLMs See Text Differently – Counting Tokens – One Token at a Time – Auto-Regressive Models – Patterns and Repetitions – Temperature and Probabilities – The Transformer Architecture.

UNIT II	RLHF, CHAT MODELS, AND DESIGNING LLM APPLICATIONS	9

Reinforcement Learning from Human Feedback: The Process of Building an RLHF Model – Keeping LLMs Honest – Avoiding Idiosyncratic Behavior – RLHF Packs a Lot of Bang for the Buck – Beware of the Alignment Tax.
Moving from Instruct to Chat: Instruct Models – Chat Models – The Changing API – Chat Completion API – Comparing Chat with Completion Models.
Designing an LLM Application: From Problem to Application – The LLM Application Design Process – Choosing the Right Model – Defining the Task – Iteration and Refinement.

UNIT III	CONTEXT, ASSEMBLING THE PROMPT, AND TAMING THE MODEL	9

Context: Sourcing Context Information – Ranking Context by Importance – Selecting Relevant Context for the Task – Few-Shot Learning – Chain-of-Thought Prompting.
Assembling the Prompt: Markdown Formatting – Snippet Prioritization – Avoiding Token Limits – Prompt Templates – Organizing Prompt Elements – Structuring for High-Quality Completions.
Taming the Model: Logprobs – Temperature Tuning – Mitigating Repetition – Steering Completions – Handling Off-Topic Tangents – Controlling Output Format and Length.

UNIT IV	CONVERSATIONAL AGENCY AND LLM WORKFLOWS	9

Conversational Agency: Designing Stateful Chatbots – Tool Integration – APIs and Databases – ReAct: Reasoning and Acting – Building Agents with Memory – Managing Conversation History.
LLM Workflows: Orchestrating Multi-Step Processes – Pipelines and Workflow Design – Task Decomposition – Chaining LLM Inferences – Error Handling in Workflows – Practical Workflow Examples.

UNIT V	EVALUATING LLM APPLICATIONS AND LOOKING AHEAD	9

Evaluating LLM Applications: Offline Evaluation – Building Example Suites – Code Correctness Testing – Online Evaluation – A/B Testing – User Engagement Metrics – Direct Feedback Data – Tracking Incidental Metrics – Using Evaluation Data for Fine-Tuning.

Looking Ahead: Multimodality – User Interfaces for LLM Applications – Ethical Considerations – Core Principles for Effective Prompt Engineering – Future Trends in LLM Applications.

TOTAL: 45 PERIODS

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

CO1: Explain the architecture of large language models including tokenization, auto-regressive generation, hallucinations, and the transformer mechanism. *[K2 – Understand]*

CO2: Describe the RLHF process, the alignment tax, and the differences between instruct and chat models. *[K2 – Understand]*

CO3: Apply context sourcing, few-shot learning, chain-of-thought prompting, and prompt assembly techniques to build effective prompts. *[K3 – Apply]*

CO4: Analyze and compare prompt strategies using logprobs, temperature tuning, and structured output techniques to optimize LLM responses. *[K4 – Analyze]*

CO5: Design conversational agents with tool integration and orchestrate multi-step LLM workflows for real-world applications. *[K6 – Create]*

CO6: Evaluate LLM applications using offline example suites, online A/B testing, and user engagement metrics with appropriate evaluation frameworks. *[K5 – Evaluate]*

TEXT BOOKS:

2. John Berryman and Albert Ziegler, "Prompt Engineering for LLMs: The Art and Science of Building Large Language Model–Based Applications", O'Reilly Media, 2025. ISBN: 9781098156138.

REFERENCES:

6. James Phoenix and Mike Taylor, "Prompt Engineering for Generative AI: Future-Proof Inputs for Reliable AI Outputs", O'Reilly Media, 2024. ISBN: 9781098153434.
7. DAIR.AI, "Prompt Engineering Guide". Freely available at https://www.promptingguide.ai
8. NPTEL Course: "Introduction to Large Language Models (LLMs)" by Prof. Tanmoy Chakraborty, IIT Delhi. Available at https://onlinecourses.nptel.ac.in/noc25_cs45
9. Hugging Face Documentation: Transformers and PEFT. Freely available at https://huggingface.co/docs
10. Anthropic, "Prompt Engineering Documentation". Freely available at https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering

24AM004	INTRODUCTION TO REINFORCEMENT LEARNING	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To know about the concepts of Reinforcement Learning
- To understand Monte Carlo Decision Process and Dynamic Programming.
- To understand temporal difference learning.
- To know how functional approximation is used in reinforcement learning.
- To study basics of Deep Reinforcement Learning.

UNIT I	INTRODUCTION TO REINFORCEMENT LEARNING	9

Introduction - Elements of RL, History of RL- Limitation and Scope - Examples – Multi-arm Bandits – n-armed Bandit Problem – Action-Value Methods – Incremental Implementation – Nonstationary Problem – Optimistic Initial Values – Upper Confidence Bound Action Selection – Gradient Bandits – Contextual Bandits.

UNIT II	MONTE DECISION PROCESS AND DYNAMIC PROGRAMMING	9

Finite Markov Decision Processes - The Agent Environment interface - Goals and Rewards – Returns – Episodic and Continuing Tasks Markov Property – MDP - Value functions – Optimality and Approximation - Dynamic Programming - Policy Evaluation - Policy Improvement, Iteration - Value Iteration - Asynchronous DP- Efficiency of DP - Monte Carlo Prediction - Monte Carlo Estimation of Action Values - Monte Carlo Control- Off - policy Monte Carlo Prediction – Off-Policy Monte Carlo Controls.

UNIT III	TEMPORAL DIFFERENCE LEARNING	9

Temporal-Difference prediction – Advantages of TD Prediction Methods -Optimality of TD(0) – Sarsa: On-policy TD Control – Q Learning: Off-Policy TD Control – n-step TD Prediction – Forward view – Backward view – SARSA – Watkins's Q – Off-policy Eligibility Traces.

UNIT IV	FUNCTION APPROXIMATION METHODS	9

Value Prediction with function Approximation – Gradient Descent Methods – Linear Methods – Control with Function Approximation – Off-Policy Approximation of Action Values – Policy Approximation – Actor-Critic Methods – Eligibility Traces – R-Learning and the Average-Reward Setting.

UNIT V	DEEP REINFORCEMENT LEARNING	9

Deep Q-Learning – Rainbow DQN – DQN Improvements – Policy Gradient Methods – Benefits – Calculation – Theorem – Policy Functions – Implementation -Hierarchical RL – Multi-Agent RL.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Learn about the concepts of Reinforcement Learning.
CO2: Understand the concept of Dynamic Programming and Monte Carlo Decision Process
CO3: Understand the concept of Temporal difference Learning (TML)
CO4: Describe how functional approximation is used in reinforcement learning.
CO5: Learn the basics of Deep Reinforcement Learning.

TEXT BOOKS:

3. Sutton R. S. and Barto A. G., "Reinforcement Learning: An Introduction", MIT Press, Second Edition, 2020.
4. Phil Winder, "Reinforcement Learning: Industrial Applications of Intelligent Agents". Oreilly, 2021.

REFERENCES:

6.Kevin Murphy, "Machine Learning - A Probabilistic Perspective" , MIT press, 2012.
7.Christopher Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.

24AM005	INTRODUCTION TO GENERATIVE AI	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To understand the basic concepts of Generative AI.
- To build Generative AI systems to generate images.
- To understand the concept used in Generative AI Models.
- To use various Generative AI models.
- To compare and use the various Large Language Models.
- To understand the basics of Prompt Engineering.

UNIT I	INTRODUCTION	9

Generative Models – Image transformation – Challenges - Deep Neural Networks – Perceptron – back propagation – CNN – RNN – Optimizer.

UNIT II	IMAGE GENERATION	9

Creating encodings of images – variational objective – Inverse Autoregressive flow – Importing CIFAR – Creating the network from TensorFlow 2.

UNIT III	GENERATIVE ADVERSARIAL NETWORKS	9

Generative Adversarial Networks – Vanilla GAN – Improved GANs – Progressive GAN – Challenges – Paired style transfer – Unpaired style transfer – Deepfakes – Modes of operation – key feature set – High level flow – Replacement – Re-enactment.

UNIT IV	LARGE LANGUAGE MODELS	9

Overview of LLMs - Transformers – GPT – Types of LLMs – Key concepts – other Transformers – T5 – Generative Pre-Training Models – Multi-modal Models – DALL.E 2

UNIT V	PROMPT ENGINEERING	9

Basics – In-Context Learning – In-Context Prompting – Techniques – Image Prompting – Prompt Hijacking – Challenges.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Elaborate the basic concepts of Generative AI.
CO2: Build Generative AI systems to generate images.
CO3: Apply the concepts used in Generative AI Models.
CO4: Use various Generative AI models.
CO5: Compare and use the various Large Language Models.
CO6: Analyze the basics of Prompt Engineering.

TEXT BOOKS:

1. Ben Auffarth, Generative AI with Lang Chain, Packt Publishing, 2023.
2. Amit Bahree, Generative AI in Action, Manning Publication, First Edition, 2023.

REFERENCES:

1. David Foster, Generative Deep Learning, 2nd Edition, O'Reilly Media, 2023.
2. Numa Dhamani and Maggie Engler, Introduction to Generative AI, Manning Publication, First Edition, 2024.
3. Valentina Alto, Modern Generative AI with ChatGPT and OpenAI Models, Packt publications, 2024.

24AM006	AGENTIC AI FOR ENGINEERS	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

1. Understand the fundamentals of Large Language Models, transformer architecture, and their role as the engine of AI agents.
2. Learn the brain, perception, and action paradigm of AI agents and explore principal agent frameworks.
3. Apply Retrieval Augmented Generation techniques to build knowledge-grounded AI agents that prevent hallucinations.
4. Design and implement knowledge graphs and GraphRAG systems for structured reasoning in AI agents.
5. Build and orchestrate single-agent and multi-agent systems for solving complex real-world tasks.

UNIT I	FOUNDATIONS OF LARGE LANGUAGE MODELS AND TEXT PROCESSING	9

Representing text for AI – One-hot encoding – Bag-of-words – TF-IDF – Embedding, application, and representation – Word2Vec – A notion of similarity for text – Properties of embeddings. The Transformer: Exploring attention and self-attention – Introducing the transformer model – Training a transformer – Exploring masked language modeling – Visualizing internal mechanisms – Applying a transformer. Exploring LLMs as a powerful AI engine – Discovering the evolution of LLMs – The scaling law – Emergent properties – Instruction tuning, fine-tuning, and alignment – Exploring smaller and more efficient LLMs – Exploring multimodal models – Understanding hallucinations and ethical and legal issues – Prompt engineering.

UNIT II	AI AGENTS – ARCHITECTURE AND FRAMEWORKS	9

Understanding the brain, perception, and action paradigm – The brain – The perception – Action. Classifying AI agents. Understanding the abilities of single-agent and multiple-agent systems. Exploring the principal libraries – LangChain – Haystack – LlamaIndex – Semantic Kernel – AutoGen – Choosing an LLM agent framework. Creating an agent to search the web. Exploring naïve RAG – Retrieval, optimization, and augmentation – Chunking strategies – Embedding strategies – Embedding databases – Evaluating the output – Comparison between RAG and fine-tuning – Using RAG to build a movie recommendation agent.

UNIT III	ADVANCED RAG AND KNOWLEDGE GRAPHS FOR AI AGENTS	9

Advanced RAG Techniques for Information Retrieval and Augmentation: Discussing naïve RAG issues – Exploring the advanced RAG pipeline – Hierarchical indexing – Hypothetical questions and HyDE – Context enrichment – Query transformation – Keyword-based search and hybrid search – Query routing – Reranking – Response optimization – Modular RAG and its integration with other systems. Creating and Connecting a Knowledge Graph to an AI Agent: Introduction to knowledge graphs – A formal definition of graphs and knowledge graphs – Taxonomies and ontologies – Creating a knowledge graph with your LLM – Knowledge creation – Knowledge assessment – Knowledge cleaning – Knowledge enrichment – Retrieving information with a knowledge graph and an LLM – GraphRAG applications.

UNIT IV	REINFORCEMENT LEARNING AND AUTONOMOUS AGENTS	9

Introduction to reinforcement learning – The multi-armed bandit problem – Markov decision processes – Deep reinforcement learning – Model-free versus model-based approaches – On-policy versus off-policy methods – Exploring deep RL in detail – Challenges and future direction for deep RL. LLM interactions with RL models – RL-enhanced LLMs – LLM-enhanced RL. Introduction to autonomous agents – Toolformer – HuggingGPT – Multi-agent for law – Multi-agent for healthcare applications.

UNIT V	MULTI-AGENT SYSTEMS AND DEPLOYMENT	9

Creating single- and multi-agent systems – Working with HuggingGPT – Using HuggingGPT locally – Using HuggingGPT on the web – Multi-agent system – SaaS, MaaS, DaaS, and RaaS – A comparison of the different paradigms. Building an AI agent application – Introduction to Streamlit – Starting with

Streamlit – Developing our frontend with Streamlit – Creating a dynamic app – Creating an application with Streamlit and AI agents – Machine learning operations and LLM operations – Model development – Model training – Model testing – Handling errors in production – Security considerations for production. The Future Ahead: AI agents in healthcare – AI agents in other sectors – Challenges and open questions – Ethical questions.

TOTAL: 45 PERIODS

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

CO1: Explain the architecture, training, and capabilities of transformers and Large Language Models. (K2)

CO2: Apply prompt engineering and text representation techniques to build LLM-based applications. (K3)

CO3: Design and implement RAG pipelines and knowledge graphs to ground AI agents in factual knowledge. (K5)

CO4: Analyze and compare AI agent frameworks, architectures, and multi-agent paradigms for given problem scenarios. (K4)

CO5: Develop autonomous single-agent and multi-agent systems that integrate planning, tool use, and knowledge retrieval. (K5)

CO6: Evaluate AI agent applications for production readiness, addressing reliability, security, and ethical considerations. (K5)

TEXT BOOKS:

1. Salvatore Raieli and Gabriele Iuculano, “Building AI Agents with LLMs, RAG, and Knowledge Graphs,” Packt Publishing, 2025.

REFERENCES:

1. Valentina Alto, “Building LLM Powered Applications: Create Intelligent Apps and Agents with Large Language Models,” Packt Publishing, 2024.
2. Roberto Infante, “AI Agents and Applications: With LangChain, LangGraph and MCP,” Manning Publications, 2025.
3. Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach,” 4th Edition, Pearson, 2022.
4. Andrew Ng, “Agentic AI,” DeepLearning.AI (Online Course), 2025.
5. OpenAI, “A Practical Guide to Building Agents,” 2025. [Available Online]

MINOR DEGREE IN ARTIFICIAL INTELLIGENCE

24AM007	DATA SCIENCE FOR ENGINEERS	L	T	P	C
		3	0	0	3

OBJECTIVES:

The Course will enable learners to:

- To learn the fundamentals of Data Science.
- To describe and understand data for analysis.
- To describe and analyze relationships between variables.
- To apply various data pre-processing strategies.
- To apply data wrangling on data for further processing.

UNIT I	INTRODUCTION	9

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – data preparation - Exploratory Data analysis – build the model – presenting findings and building applications - Data Mining - Data Warehousing – Basic statistical descriptions of Data.

UNIT II	DESCRIBING DATA	9

Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores.

UNIT III	DESCRIBING RELATIONSHIPS	9

Correlation –Scatter plots –correlation coefficient for quantitative data –computational formula for correlation coefficient – Regression –regression line –least squares regression line –
Standard error of estimate – interpretation of r2 –multiple regression equations –regression towards the mean.

UNIT IV	DATA PREPROCESSING	9

Data Preprocessing – Purpose – Tools – Levels – Cleaning up the Table – Unpacking – Restructuring – Reformulating the Table - Data Cleaning: – Missing Values – Outliers – Errors - Data Fusion vs Data Integration – Directions – Adding attributes – data objects - Entity identification – Data Reduction vs data redundancy – Types – Performing Numerosity Data reduction – Sampling – Principal Component Analysis – Data Transformation – Normalization and Standardization – Discretization – Smoothing – Aggregation - Binning.

UNIT V	DATA WRANGLING	9

Processing Uni-dimensional Data: Creating Vectors - Inspecting the Data Distribution with Histograms- Aggregating Numerical Data - Arithmetic Operators - Indexing vectors- Multidimensional data: Creating matrices - Reshaping - visualization (2d,3d) - Heterogeneous Data: Data frames - Aggregation - Transformation - Indexing - Accessing Database: Filtering - Ordering - Removing Duplicates - Grouping and Aggregating - Joining - Handling with Many Files.

TOTAL:45 PERIODS

OUTCOMES:

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

CO1: Explain the fundamentals of data science.
CO2: Illustrate the basics of data for analysis.
CO3: Identify relationships between variables.
CO4: Implement various data pre-processing strategies.
CO5: Implement data wrangling for further processing of data.
CO6: Apply Data preprocessing on real-time data sets.

TEXT BOOKS:

1. Robert S. Witte and John S. Witte, “Statistics”, Eleventh Edition, Wiley Publications, 2017. (Units 2, 3)
2. Roy Jafari, Hands-On Data Preprocessing in Python: Learn how to effectively prepare data for successful data analytics, Packt Publications, First Edition, 2022. (Unit 4)

3. Marek Gagolewski, Minimalist Data Wrangling with Python, Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0), 2022. (Unit 5)

REFERENCES:

1. David Cielen, Arno D. B. Meysman, and Mohamed Ali, “Introducing Data Science”, Manning Publications, 2016. (Unit 1)
2. Hui Lin, Ming Li, Practitioner’s Guide to Data Science, 1st Edition, Chapman and Hall/CRC, 2023.
3. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann, 2012.
4. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data, Packt Publications, 2020.
5. Dr. Tirthajyoti Sarkar, Shubhadeep Roychowdhury, Data Wrangling with Python: Creating actionable data from raw sources, Packt Publications, First Edition, 2019.
6. Robert H. Shumway, David S. Stoffer, Time Series Analysis and Its Applications With R Examples, Fourth Edition, 2016, Springer.
7. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements Of Statistical Learning: Data Mining, Inference, and Prediction, Springer, Second Edition, 2017.
8. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, Third Edition,2022.
9. Jake VanderPlas, “Python Data Science Handbook – Essential tools for working with data”, O’Reilly, 2017.
10. Ashwin Pajankar, Aditya Joshi, Hands-on Machine Learning with Python: Implement Neural Network Solutions with Scikit-learn and PyTorch, Apress, 2022.
11. NPTEL Courses:
    a. Data Science for Engineers - https://onlinecourses.nptel.ac.in/noc23_cs17/preview
    b. Python for Data Science - https://onlinecourses.nptel.ac.in/noc23_cs21/preview

24AM008	MACHINE LEARNING ALGORITHMS	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To discuss the basics of Machine Learning and model evaluation.
- To study dimensionality reduction techniques.
- To understand the various classification algorithms.
- To elaborate on unsupervised learning techniques.
- To discuss the basics of neural networks and various types of learning.

UNIT I	INTRODUCTION	9

Machine Learning – Types – Applications – Preparing to Model – Activities – Data – Exploring structure of Data – Data Quality and Remediation – Data Pre-processing – Modelling and Evaluation: Selecting a Model -Training a Model – Model representation and Interpretability – Evaluating Performance of a Model – Improving Performance.

UNIT II	FEATURE ENGINEERING AND DIMENSIONALITY REDUCTION	9

Feature Engineering – Feature Transformation – Feature Subset Selection - Principle Component Analysis – Feature Embedding – Factor Analysis – Singular value decomposition and Matrix Factorization – Multidimensional scaling – Linear Discriminant Analysis – Canonical Correlation Analysis – Isomap – Locally linear Embedding – Laplacian Eigenmaps.

UNIT III	SUPERVISED LEARNING	9

Linear Regression -Relation between two variables – Steps – Evaluation – Logistic Regression – Decision Tree – Algorithms – Construction – Classification using Decision Tree – Issues – Rule-based Classification – Pruning the Rule Set – Support Vector Machines – Linear SVM – Optimal Hyperplane – Radial Basis Functions – Naïve Bayes Classifier – Bayesian Belief Networks.

UNIT IV	UNSUPERVISED LEARNING	9

Clustering – Types – Applications - Partitioning Methods – K-means Algorithm – K-Medoids – Hierarchical methods – Density based methods DBSCAN – Finding patterns using Association Rules – Hidden Markov Model.

UNIT V	NEURAL NETWORKS AND TYPES OF LEARNING	9

Biological Neuron – Artificial Neuron – Types of Activation function – Implementations of ANN – Architectures of Neural Networks – Learning Process in ANN – Back propagation – Deep Learning – Representation Learning – Active Learning – Instance based Learning – Association Rule Learning – Ensemble Learning Algorithm: Bagging – Boosting – GBM – XGBoost – Regularization Algorithm-Reinforcement Learning – Elements.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Explain the basics of Machine Learning and model evaluation.
CO2: Study dimensionality reduction techniques.
CO3: Understand and implement various classification algorithms.
CO4: Understand and implement various unsupervised learning techniques.
CO5: Build Neural Networks and understand the different types of learning.
CO6: Use Machine Learning Algorithms to build applications.

TEXT BOOKS:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, “Machine Learning”, Pearson, 2019. (Unit 1 – chap 1,2,3/ Unit 2 – Chap 4 / Unit 4 – 9 / Unit 5 – Chap 10, 11)
2. Ethem Alpaydin, “Introduction to Machine Learning, Adaptive Computation and Machine Learning Series”, Third Edition, MIT Press, 2014. (Unit 2 – Chap 6 / Unit 4 – chap 8.2.3/ Unit 5 – Chap 18)

REFERENCES:

1. Anuradha Srinivasaraghavan,Vincy Joseph, “Machine Learning”, First Edition, Wiley, 2019.(Unit 3 – Chap 7,8,9,10,11 / Unit 4 – 13, 11.4, 11.5,12)
2. Sebastian Raschka, Python Machine Learning, Packt Publications, Second Edition, 2017.

3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly Publications, Third Edition, 2022.
4. Peter Harrington, “Machine Learning in Action”, Manning Publications, 2012.
5. Stephen Marsland, “Machine Learning – An Algorithmic Perspective”, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
6. Tom M Mitchell, “Machine Learning”, First Edition, McGraw Hill Education, 2013.
7. Bishop C.M., Pattern Recognition And Machine Learning, Springer, 2009.
8. NPTEL Courses:
    a. Introduction to Machine Learning - https://onlinecourses.nptel.ac.in/noc23_cs18/preview

24AM009	DEEP LEARNING TECHNIQUES	L	T	P	C
		3	0	0	3

OBJECTIVES:

- To outline the basics of deep neural networks.
- To discuss advanced deep learning models.
- To discuss CNN and RNN architectures of deep neural networks.
- To elaborate autoencoders in neural networks.
- To discuss the deep generative models.

UNIT I	DEEP NETWORKS	9

Challenges motivating deep learning - Deep feedforward networks - Learning XOR - Gradient based learning - Hidden Units – Architecture Design – Back Propagation – Regularization – Parameter Norm Penalties – Constrained Optimization – Under-Constrained Problems – Dataset Augmentation – Noise Robustness – Semi-Supervised Learning – Multi-Task Learning – Early Stopping – Parameter Tying and Sharing – Bagging and Other Ensemble methods – Dropout – Adversarial Training.

UNIT II	OPTIMIZATION FOR TRAINING DEEP MODELS	9

Pure optimization – Challenges – Basic Algorithms – Parameter initialization Strategies – Algorithms with Adaptive Learning Rates – Approximate Second-Order methods – Optimization Strategies and Meta Algorithms.

UNIT III	CONVOLUTIONAL AND RECURRENT NEURAL NETWORKS	9

Convolution Operation – motivation – Pooling – Infinitely Strong prior – Variants – Structured Output – Data Types – Efficient Convolutional Algorithms – Random or Unsupervised features – Neuroscientific Basis - Deep Learning – Sequence Modelling - Computational Graphs - RNN - Bidirectional RNN – Encoder-Decoder - Sequence to Sequence RNN - Deep Recurrent Networks - Recursive Neural Networks -- Long Term Dependencies; Leaky Units – Strategies for multiple time scales – LSTM and Gated RNNs – Optimization for Long Term Dependencies.

UNIT IV	AUTOENCODERS	9

Autoencoders: Undercomplete autoencoders - Regularized autoencoders – Power, Layer Size and Depth - Stochastic encoders and decoders – Denoising Autoencoders - Learning with autoencoders – contractive Autoencoders – Applications of autoencoders.

UNIT V	DEEP GENERATIVE MODELS	9

Boltzmann Machine – Restricted Boltzmann Machine – Deep Belief Networks – Deep Boltzmann Machines - Boltzmann Machines for Real-Valued Data – Convolutional Boltzmann Machines - Boltzmann Machine for Structured or Sequential Outputs – Directed Generative Nets – Evaluating Generative Models.

TOTAL: 45 PERIODS

OUTCOMES:

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

CO1: Outline the basics of deep neural networks**.**
CO2: Develop advanced deep learning models.
CO3: Implement CNN and RNN architectures of deep neural networks.
CO4: Interpret autoencoders in neural networks.
CO5: Apply deep generative models to solve real world problems.
CO6: Build deep learning models and evaluate them.

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, ``Deep Learning", MIT Press, 2016.

REFERENCES:

1. Charu C. Aggarwal, ``Neural Networks and Deep Learning: A Textbook", Springer International Publishing, 2018.
2. Yoav Goldberg, ``Neural Network Methods for Natural Language Processing", Synthesis Lectures on Human Language Technologies, Morgan & Claypool publishers, 2017.
3. Francois Chollet, ``Deep Learning with Python", Manning Publications Co, 2018.
4. Josh Patterson, Adam Gibson, ``Deep Learning: A Practitioner's Approach", O'Reilly Media,

2017.
5. Navin Kumar Manaswi, “Deep Learning with Applications Using Python”, Apress, 2018.

NPTEL:

6. Deep Learning - https://onlinecourses.nptel.ac.in/noc24_ee04/preview
7. Deep Learning - IIT Ropar - https://onlinecourses.nptel.ac.in/noc24_cs59/preview

24AM010	FOUNDATIONS OF LARGE LANGUAGE MODELS	L	T	P	C
		3	0	0	3

OBJECTIVES: The Course will enable learners to:

- Understand text preprocessing, tokenization, and vector representations used in NLP and LLMs.
- Learn the basic concepts of language models used in Large Language Models (LLMs).
- Understand Transformer architecture and Large Language Models for text generation tasks.
- Use pre-trained Hugging Face models for NLP applications and fine-tuning.
- Build LLM-powered applications using Prompt Engineering, RAG, and PEFT techniques.

UNIT I	FOUNDATIONS OF NLP AND EMBEDDINGS	9

Introduction to NLP – Words, Tokens, and Subword Tokenization – Unicode and Text Preprocessing – Byte Pair Encoding (BPE) – Corpora and Regular Expressions – Word Embeddings – Vector Semantics – Cosine Similarity – Word2Vec – Semantic Relationships in Embeddings – Bias and Evaluation of Embeddings.

UNIT II	LANGUAGE MODELS	9

Introduction to Language Models – Statistical Language Models – N-gram Language Models – Probability in NLP – Markov Assumptions – Smoothing Techniques – Neural Language Model Concepts – Continuous Bag of Words (CBOW) – Skip-Gram Model – Perplexity and Evaluation Metrics – Tokenization Strategies – Contextual Representations – Text Corpora and Vocabulary Building – Language Modeling Applications in NLP.

UNIT III	TRANSFORMERS AND LARGE LANGUAGE MODELS	9

Introduction to Large Language Models – Auto-Regressive Language Modeling – Prompting Techniques – Text Generation and Sampling – Transformer Architecture – Self-Attention – Multi-Head Attention – Positional Encoding – Decoder-Only Architectures – Training and Evaluation of LLMs – Ethical and Safety Issues.

UNIT IV	PRE-TRAINED AND MULTIMODAL MODELS	9

Masked Language Models – BERT and Contextual Embeddings – Fine-Tuning for Classification and Named Entity Recognition – Hugging Face Transformers and Trainer API – Encoder–Decoder Translation Models – Beam Search – Diffusion Models – Stable Diffusion – Text-to-Image Generation – Multimodal AI Applications

UNIT V	LLM APPLICATIONS AND OPTIMIZATION	9

Prompt Engineering – Zero-shot and Few-shot Prompting – Chain-of-Thought Prompting – Retrieval-Augmented Generation (RAG) – Dense Vector Retrieval – Vector Databases – LoRA and QLoRA – Quantization Techniques – Parameter-Efficient Fine-Tuning (PEFT) – HuggingFace Pipelines and Open-Source LLM Applications.

TOTAL: 45 PERIODS

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

CO1: Explain NLP preprocessing, tokenization, and embedding techniques used in language models.

CO2: Explain statistical and neural language modeling techniques used as foundations for Large Language Models**.**

CO3: Analyze Transformer architecture and implement Large Language Models for text generation tasks.

CO4: Use and fine-tune pre-trained HuggingFace models for NLP applications.

CO5: Develop LLM-powered systems using Prompt Engineering, RAG, and PEFT techniques.

CO6: Evaluate ethical issues such as bias, hallucination, and privacy in Large Language Models

TEXT BOOKS:

1. Daniel Jurafsky and James H. Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models", 3rd Edition, Stanford University, 2026. [Primary reference – Volume I: Large Language Models. Available free online: https://web.stanford.edu/~jurafsky/slp3/

REFERENCES:

1. Tong Xiao and Jingbo Zhu, "Foundations of Large Language Models", NLP Lab, Northeastern University and NiuTrans Research, 2025 (Creative Commons Licensed). Available free online: https://arxiv.org/abs/2501.09223
2. Sebastian Raschka, "Build a Large Language Model (From Scratch)", 1st Edition, Manning Publications, 2024. Companion code (open source): https://github.com/rasbt/LLMs-from-scratch
3. Tanmoy Chakraborty, "Introduction to Large Language Models", 1st Edition, Wiley India, 2025. ISBN: 9789363864740.
4. Jacob Eisenstein, "Natural Language Processing", 1st Edition, The MIT Press, 2019.
5. HuggingFace Team, "HuggingFace NLP Course" and "Transformers Documentation". Freely available at https://huggingface.co/learn/nlp-course