RAMDEOBABA UNIVERSITY, NAGPUR–440013

Established by the Maharashtra Private Universities (Establishement and Regulation)Act2023(Mah. ActNoVIIIof2024)

Formerly ,Shri Ramdeobaba College of Engineering and Management,Nagpur440013

School of Electrical and Electronics Engineering

Department of (Electronics Engineering)

PROGRAMME SCHEME & SYLLABI

of First Year as per National Education Policy(NEP)

(With effect from Academic Year 2024-25)

B.Tech.

ELECTRONICS AND COMPUTER SCIENCE

Vision

Department endeavors to facilitate state of the art technical education in the field of Electronics and Computer Science to produce globally competent engineering professionals.

Mission

- To impart quality education in the field of Electronics and Computer Science Engineering.
- To foster mutually beneficial relationship with industries, academics and research organizations.
- To create an intellectually stimulating environment for learning, research and innovation with professional and ethical values.

Programme Educational Objectives (PEO)

PEO1: Exercise the acquired knowledge and skills in electronics and computer science to solve engineering problems.

PEO2: Engage in professional development by pursuing higher education and research.

PEO3: Demonstrate leadership qualities, effective communication skills, and to work in a team in a diverse environment with strong adherence to professional ethics

PEO4: Engage in lifelong learning for sustained career advancement and adapt to changing professional and societal needs.

Program Outcomes (POs):

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 analyse 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)

Knowledge and Attitude Profile (WK)
WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

WK2: Conceptually-based mathematics, numerical analysis, data analysis, statistics and formal aspects of computer and information science to support detailed analysis and modelling applicable to the discipline.

WK3: A systematic, theory-based formulation of engineering fundamentals required in the engineering discipline.

WK4: Engineering specialist knowledge that provides theoretical frameworks and bodies of knowledge for the accepted practice areas in the engineering discipline; much is at the forefront of the discipline.

WK5: Knowledge, including efficient resource use, environmental impacts, whole-life cost, reuse of resources, net zero carbon, and similar concepts, that supports engineering design and operations in a practice area.

WK6: Knowledge of engineering practice (technology) in the practice areas in the engineering discipline.

WK7: Knowledge of the role of engineering in society and identified issues in engineering practice in the discipline, such as the professional responsibility of an engineer to public safety and sustainable development.

WK8: Engagement with selected knowledge in the current research literature of the discipline, awareness of the power of critical thinking and creative approaches to evaluate emerging issues.

WK9: Ethics, inclusive behavior and conduct. Knowledge of professional ethics, responsibilities, and norms of engineering practice. Awareness of the need for diversity by reason of ethnicity, gender, age, physical ability etc. with mutual understanding and respect, and of inclusive attitudes.

Programme Specific Outcomes (PSOs):

PSO1: Apply principles of**electronics, signal processing, and embedded systems**to **design, develop, and analyze** innovative solutions for complex engineering problems in **automation,** healthcare,**communication, and intelligent control systems,** while **evaluating** their performance for real-world applications.

PSO2: Apply computational techniques, algorithms, and software development methodologies **to analyze, evaluate, and create** AI-driven, cloud-based, and secure computing solutions for real-world applications.

PSO3:Design, develop, and evaluate smart and secure computing systems using AI, embedded systems, IoT, and cloud technologies **to create** solutions for communication, healthcare, agriculture, cybersecurity, and big data applications.

Semester I

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration (Hrs.)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	ESC	24EE01 TH0101	Basic Electronics and Computer Fundamentals	3	0	3	50	50	100	3
2	BSC	24HS03 TH0102	Calculus and Linear Algebra	3	0	3	50	50	100	3
3	ESC	24EE01 TH0102	Digital Logic Design	3	0	3	50	50	100	3
4	ESC	24EE01 PR0102	Digital Logic Design	0	2	1	25	25	50	-
5	ESC	24EE01 TH0103	Fundamentals of Programming	3	0	3	50	50	100	3
6	ESC	24EE01 PR0103	Fundamentals of Programming	0	2	1	25	25	50	-
7	VSEC	24EE01 PR0104	Computer workshop-I	0	2	1	25	25	50	-
8	VSEC	24EE01 PR0105	Prompt Engineering	0	2	1	25	25	50	-
9	CCA	24HS04 PR0101	Sports-Yoga-Recreation	0	2	1	25	25	50	-
10	CCA	24HS02 PR0105	Liberal/Performing Art Lab	0	2	1	25	25	50	-
11	VSEC	24EE01 TH0106	Creativity Innovation and Design Thinking	1	0	1	50	-	50	-
12	VSEC	24EE01 PR0106	Creativity Innovation and Design Thinking	0	2	1	25	25	50	-
			TOTAL	**13**	**14**	**20**				

Semester II

SN	Course Type	Code	Course	Hours/ week		C	Maximum marks			ESE Duration(Hrs.)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	PCC	24EE01TH0201	CMOS Digital Circuit Design	3	0	3	50	50	100	3
2	PCC	24EE01PR0201	CMOS Digital Circuit Design	0	2	1	25	25	50	-
3	BSC	24HS03TH0217	Probability and Statistics	3	0	3	50	50	100	3
4	PCC	24EE01TH0202	Digital System Design	3	0	3	50	50	100	3
5	PCC	24EE01PR0202	Digital System Design	0	2	1	25	25	50	-
6	ESC	24EE01TH0203	Elements of Internet of Things	3	0	3	50	50	100	3
7	ESC	24EE01PR0203	Elements of Internet of Things	0	2	1	25	25	50	-
8	PCC	24EE01TH0204	Object Oriented Programming	3	0	3	50	50	100	3
9	PCC	24EE01PR0204	Object Oriented Programming	0	2	1	25	25	50	-
10	VSEC	24EE01PR0205	Computer workshop -II	0	2	1	25	25	50	-
11	AEC	24HS02TH0201	English for Professional Communication	2	0	2	50	50	100	2
12	AEC	24HS02PR0201	English for Professional Communication	0	2	1	25	25	50	-
13	IKS	24HS02TH0203	Foundational Literature of Indian Civilization	1	0	1	50	-	50	-
			TOTAL	**18**	**12**	**24**				

Exit option: Award of UG Certificate with additional 8 credits			
Exit Courses			
1	IT Support Engineer	Online/Offline Certification Course	8
2	Web Designer		8
3	UI/UX Design		8

Semester III

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs.)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	PCC	24EE01TH0301	Data Structures and Algorithms	3	0	3	50	50	100	3
2	PCC	24EE01PR0301	Data Structures and Algorithms	0	2	1	25	25	50	-
3	PCC	24EE01TH0302	Computer Architecture	3	0	3	50	50	100	3
4	PCC	24EE01PR0302	Computer Architecture	0	2	1	25	25	50	-
5	BSC	24EE01TH0303	Discrete Mathematics	3	0	3	50	50	100	3
6	PCC	24EE01TH0304	Digital Signal Processing	3	0	3	50	50	100	3
7	PCC	24EE01PR0304	Digital Signal Processing	0	2	1	25	25	50	-
8	MDM	24EE01TH0305	MDM-1	3	0	3	50	50	100	3
9	OE	24EEOEC01TH0306	Open Elective I / MOOC	2	0	2	50	50	100	2
10	BSC	24HS01TH0301	Environmental Science	1	0	1	50	-	50	-
11	BSC	24HS01PR0301	Environmental Science	0	2	1	25	25	50	-
12	VEC	24HS02TH0301	Foundational course in Universal Human Value	1	0	1	50	-	50	-
			TOTAL	**19**	**8**	**23**				

Semester IV

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs.)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	PCC	24EE01TH0401	Embedded System Design	3	0	3	50	50	100	3
2	PCC	24EE01PR0401	Embedded System Design	0	2	1	25	25	50	-
3	PCC	24EE01TH0402	Operating System	3	0	3	50	50	100	3
4	PCC	24EE01PR0402	Operating System	0	2	1	25	25	50	-
5	PCC	24EE01TH0403	Design and Analysis of Algorithms	3	0	3	50	50	100	3
6	PCC	24EE01TH0404	Fundamentals of AI and Machine Learning	3	0	3	50	50	100	3
7	PCC	24EE01PR0404	Fundamentals of AI and Machine Learning	0	2	1	25	25	50	-
8	MDM	24EE01TH0405	MDM-2	3	0	3	50	50	100	3
9	OE	24EEOEC01TH0406	Open Elective-II/MOOCs	2	0	2	50	50	100	2
10	VSEC	24EE01PR0407	Software Laboratory Practice-I	0	2	1	50	-	50	-
11	AEC	24EE01PR0408	Basic Competitive coding	0	2	1	50	50	100	-
12	HSSM	24SM01TH0401	Innovations and Entrepreneurship	1	0	1	50	0	50	-
	HSSM	24SM01PR0401	Innovations and Entrepreneurship	0	2	1	25	25	50	-
			TOTAL	**18**	**12**	**24**				

Exitoption :Award of UG Diploma with additional 8 credits			
Exit Course			
1	Android Application Development	Online/ OfflineCertification Course	8
2	Python Programming		8
3	PCB Design		8

Semester V

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs.)
				L	P		Continuous Evaluation	End Sem / Internal Eval	Total	
1	PCC	24EE01TH0501	Computer Networks	3	0	3	50	50	100	3
2	PCC	24EE01PR0501	Computer Networks	0	2	1	25	25	50	-
3	PCC	24EE01TH0502	Digital Image Processing	3	0	3	50	50	100	3
4	PCC	24EE01PR0502	Digital Image Processing	0	2	1	25	25	50	-
5	PCC	24EE01TH0503	Deep Learning	3	0	3	50	50	100	3
6	PCC	24EE01PR0503	Deep Learning	0	2	1	25	25	50	-
7	OE	24EEOEC01TH0504	Open Elective-III/MOOCs	2	0	2	50	50	100	2
8	MDM	24EE01TH0505	MDM-3	3	0	3	50	50	100	3
9	PEC	24EE01TH0506	Programme Elective-I	3	0	3	50	50	100	3
10	PEC	24EE01PR0506	Programme Elective-I	0	2	1	25	25	50	-
11	AEC	24HS02TH0501	Business Communication	1	0	1	50	0	50	-
12	AEC	24HS02PR0501	Business Communication	0	2	1	25	25	50	
			TOTAL	**18**	**10**	**23**				

Semester VI

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	PCC	24EE01TH0601	Database Management System	3	0	3	50	50	100	3
2	PCC	24EE01PR0601	Database Management System	0	2	1	25	25	50	-
3	PCC	24EE01TH0602	Software Engineering	3	0	3	50	50	100	3
4	PEC	24EE01TH0603	Programme Elective-II	3	0	3	50	50	100	3
5	PEC	24EE01PR0603	Programme Elective-II	0	2	1	25	25	50	-
6	PEC	24EE01TH0604	Programme Elective-III	3	0	3	50	50	100	3
7	PEC	24EE01PR0604	Programme Elective-III	0	2	1	25	25	50	-
8	MDM	24EE01TH0605	MDM-4	3	0	3	50	50	100	3
9	VSEC	24EE01PR0606	Software Laboratory Practice -II	0	2	1	25	25	50	-
10	AEC	24EE01PR0607	Advanced Competitive coding	0	2	1	25	25	50	-
11	FP/CEP	24EE01PR0608	Project-I	0	4	2	50	50	100	-
			TOTAL	**15**	**14**	**22**				

Exitoption: Award of UG Degree with additional 8 credits			
Exit Course			
1	TBI/Industry/Research Internship	Online/offline Certification Course	8
2	AI & ML on Cloud Platform		8
3	HLS Design		8

Semester VII/VIII

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs)
				L	P		Continuous Evaluation	End Sem/ InternalEval	Total	
1	PCC	24EE01TH0701	Fundamentals of Digital Communication	3	0	3	50	50	100	3
2	PCC	24EE01PR0701	Fundamentals of Digital Communication	0	2	1	25	25	50	-
3	PEC	24EE01TH0702	Programme Elective-IV	3	0	3	50	50	100	3
4	VEC	24EE01TH0703	Cyber Laws and Ethics in IT	2	0	2	50	50	100	2
5	PRJ	24EE01PR0704	Project-II	0	8	4	50	50	100	-
6	FP	24EE01PR0705	Internship Evaluation	0	2	0	-	-		-
7	AEC	24EE01PR0706	Participative Learning	0	2	1	25	25	50	-
			TOTAL	**8**	**14**	**14**				

Semester VIII/VII

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration (Hrs)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	PEC	24EE01TH0801	Programme Elective-V	3	0	3	50	50	100	3
2	PEC	24EE01TH0802	Programme Elective-VI	3	0	3	50	50	100	3
3	PRJ	24EE01PR0803	Project-III	0	12	6	50	50	100	-
			TOTAL	**6**	**12**	**12**				

OR

SN	Course Type	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	Internship /OJT	24EE01PR0804	Industry Internship/ TBI Internship/ Research Internship*	0	24	12	50	50	100	-
			TOTAL	0	24	**12**				

HONORS Specialization in Research

SN	Sem	Code	Course	Hours/week		C	Maximum marks			ESE Duration(Hrs)
				L	P		Continuous Evaluation	End Sem/ Internal Eval	Total	
1	VII	24EE01HT0701	Research Methodology	3	0	3	50	50	100	3
2	VII	24EE01HP0702	Research Project Phase -I	0	6	3	50	50	100	-
3	VIII	24EE01HP0801	Research Project Phase-II	0	24	12	50	50	100	-
			TOTAL	**3**	**30**	**18**				
				33 Hrs.						

Programme Electives

	Semester V	Semester VI		Semester VII	Semester VIII	
	Elective-I	**Elective-II**	**Elective-III**	**Elective-IV**	**Elective-V**	**Elective-VI**
Course Code	24EE01TH0506-1/24EE01PR0506-1	24EE01TH0603-1/24EE01PR0603-1	24EE01TH0604-1/24EE01PR0604-1	24EE01TH0702-1	24EE01TH0801-1	24EE01TH0802-1
Course Name	Deep Learning for Visual Recognition	AI & ML on Cloud Platform	Natural Language Processing	Embedded Machine Learning	Generative AI-I	Generative AI-II
Course Code	24EE01TH0506-2/24EE01PR0506-2	24EE01TH0603-2/24EE01PR0603-2	24EE01TH0604-2/24EE01PR0604-2	24EE01TH0702-2	24EE01TH0801-2	24EE01TH0802-2
Course Name	FPGA based System Design	Digital Design Verification	Design for Testability	C Based VLSI Design	VLSI Signal Processing	VLSI Physical Design
Course Code	24EE01TH0506-3/24EE01PR0506-3	24EE01TH0603-3/24EE01PR0603-3	24EE01TH0604-3/24EE01PR0604-3	24EE01TH0702-3	24EE01TH0801-3	24EE01TH0802-3
Course Name	System on Chip Design	Embedded Linux	Real-Time Operating Systems Design and Programming	Internet of Things	Mechatronics and Robotics	Graphics and Mobile Gaming
Course Code	24EE01TH0506-4/24EE01PR0506-4	24EE01TH0603-4/24EE01PR0603-4	24EE01TH0604-4/24EE01PR0604-4	24EE01TH0702-4	24EE01TH0801-4	24EE01TH0802-4
Course Name	Fundamentals of Cloud Computing	Fundamentals of Cloud Security	Parallel Architecture	Distributed Computing	Cloud Services and AI Interpretation	Edge AI Computing

HONORS Specialization in Full-Stack JavaScript Developer

SN	Sem	Code	Course	Hours/ week				Continuous Evaluation	End Sem Exam	Total Marks	ESE duration(Hrs.)
				L	T	P	C				
1	III	24EE01HT0301	JavaScript Programming Essentials	3	0	0	3	50	50	100	3
2	IV	24EE01HT0401	Front and Back end App development	3	0	0	3	50	50	100	3
3	V	24EE01HT0501	Cloud native applications, Containers and Microservices	3	1	0	4	50	50	100	3
4	VI	24EE01HT0601	Back-end Database Applications and JavaScript Full Stack Capstone Project	3	1	0	4	50	50	100	3
5	VII	24EE01HP0701	Capstone Project	0	0	8	4	50	50	100	
TOTAL				**12**	**2**	**8**	18				

MINOR Specialization in IoT

SN	Sem	Code	Course	Hours/ week				Continuous Evaluation	End Sem Exam	Total Marks	ESE duration (Hrs.)
				L	T	P	C				
1	III	24EE01MT0301	IoT fundamentals	3	0	0	3	50	50	100	3
2	IV	24EE01MT0401	Sensor Interfacing with Arduino and ESP8266	3	0	0	3	50	50	100	3
3	V	24EE01MT0501	Cloud Computing Using Raspberry Pi	3	1	0	4	50	50	100	3
4	VI	24EE01MT0601	Data Management and Analytics for IoT	3	1	0	4	50	50	100	3
5	VII	24EE01MP0701	Minor Project	0	0	8	4	50	50	100	
TOTAL				**12**	**2**	**8**	**18**				

List of MDM

Multidisciplinary Minor (MDM) Track-1: Integrated Circuit Design (IC design)

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

Multidisciplinary Minor (MDM) Track-2: IoT for Environmental Sustainability

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

Course Code	24EE01TH0101			
Category	Engineering Science Course			
Course Title	Basic Electronics and Computer Fundamentals			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	I

Course Outcomes

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

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

Syllabus

Module I:

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

Module II:

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

Module III

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

Module IV

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

Module V

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

Textbooks

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

Re**ference Books**

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

Course Code	24HS03TH0102			
Category	Basic Science Course			
Course Title	Calculus and Linear Algebra			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	I

CourseOutcomes

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

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

Syllabus

Module 1 : (8 Lectures)

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

Module 2:(10 Lectures)

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

Module 3:(8 Lectures)

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

Module 4:(8 Lectures)

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

Module 5:(6 Lectures)

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

Textbooks/References

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

Course Code	24EE01TH0102/24EE01PR0102			
Category	Engineering Science Course			
Course Title	Digital Logic Design			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	I

Course Outcomes

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

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

Module-I

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

Module-II

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

Module-III

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

Module-IV

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

Module-V

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

Module-VI

Design of synchronous sequential circuit using Mealy model and Moore model

Textbooks:

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

ReferenceBooks:

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

List of Experiments

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

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

Course Code	24EE01TH0103/24EE01PR0103			
Category	Engineering Science Course			
Course Title	Fundamentals of Programming			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	I

Course Outcomes

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

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

Syllabus

Module1 Introduction to Programming

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

Module2 C Programming Language

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

Module3 Arrays and Basic Algorithms

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

Module4 Functions and Recursion

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

Module5 Pointers and Structures

Structures, Defining structures, Array of Structures, Introduction to pointers, Defining pointers, Pointer

arithmetic, pointer operators, Use of Pointers in self-referential structures.

Module6 File handling

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

Text Books

1. Programming in ANSIC: E.Balguruswami McGraw Hill
2. Mastering C:K.R.VenugopalandS.R.Prasad,TataMcGrawHill

Reference Books

1. Programming with C: Byron Gottfried, Schaums Outline Series.
2. LetUsC: Yashwant Kanetkar, BPB Publication

List of Experiments

1.Using basic data types of C, implement arithmeticexpressions.2.Implement Programs using Decision Control Structures

3. Demonstrate use of Loop Control Structures

4.Implement Programs using Multi-way Decision Control Structures (Switch Case)

5. Apply Functions and Recursion to simply programs.

6.Initializearrayandapplyitto solve problems of 1Dand2Darrays.

7.DemonstrateuseofStructuresandPointers.8. Apply file handling concepts in C.

Note:-2/3PracticeProgramswill be taken on each of the experiments mentioned above.

Course Code	24EE01PR0104			
Category	Vocational and Skill Enhancement Course			
Course Title	Computer workshop-I			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Advanced Excel Experiments:

1. **Experiment 1: Data Analysis using Formulas**

Objective: To understand and apply various Excel formulas for data analysis.

- **Tasks:**
  - Use **SUM**, **AVERAGE**, **MAX**, **MIN**, and **IF** functions to analyze a dataset (e.g., sales data).
  - Calculate the total sales, average sales, maximum and minimum sales values.
  - Use the **IF** function to categorize data (e.g., sales above a certain threshold as "High" and below as "Low").

2. **Experiment 2: Data Visualization and Filtering**
   - **Objective:** To create and format graphs and apply filters to a dataset.
   - **Tasks:**
     - Plot a **line graph** and **bar chart** using given data (e.g., monthly expenses).
     - Apply **filters** to display specific data, such as expenses in a particular month or category.
     - Customize the chart with titles, labels, and colors.

HTML, CSS, and JavaScript Experiments:

1. **Experiment 1: Basic HTML Structure**
   - **Objective:** To create a simple webpage using basic HTML elements.
   - **Tasks:**
     - Create an HTML page with a **title**, **heading**, **paragraph**, **list** (ordered and unordered), and **image**.
     - Structure the content using ** ** and **** tags.
     - Use **<a>** tags to create hyperlinks.
2. **Experiment 2: Styling with CSS**
   - **Objective:** To apply basic CSS to style an HTML page.
   - **Tasks:**
     - Create an HTML page and link an external CSS file.
     - Use CSS to set the **background color**, **font style**, **text alignment**, and **margins**.
     - Apply CSS to style **buttons** and **images** (e.g., rounded corners, hover effects).

3. **Experiment 3: Creating a Simple Web Form**
    - **Objective:** To design a basic web form using HTML and CSS.
    - **Tasks:**
        - Create a web form with fields like **text input**, **password**, **email**, **dropdown menu**, **radio buttons**, and **submit button**.
        - Style the form using CSS to improve its appearance (e.g., padding, border-radius, hover effects).
        - Validate the form fields using basic HTML5 attributes (e.g., **required**).
4. **Experiment 4: Introduction to JavaScript**
    - **Objective:** To add interactivity to a webpage using JavaScript.
    - **Tasks:**
        - Create an HTML page with a **button** that, when clicked, displays an **alert** message.
        - Write a simple JavaScript function to **change the content** of a paragraph when a button is clicked.
        - Use JavaScript to perform basic arithmetic operations (e.g., adding two numbers) and display the result on the page.

Course Code	24EE01PR0105			
Category	Vocational and Skill Enhancement Course			
Course Title	Prompt Engineering			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Module 1: Introduction to Prompt Engineering

- Lab 1: What is Prompt Engineering, and why do we care?
- Lab 2: English as a new programming language

Module 2: Getting Started with Prompt Engineering

- Lab 3: Getting to know our GPT-based AI tool
- Lab 4: The Naive Prompting Approach and the Persona Pattern
- Lab 5: The Interview Pattern

Module 3: The Chain-of-Thought Approach

- Lab 6: The Chain-of-Thought Approach in Prompt Engineering

Module 4: Advanced Techniques

- Lab 7: The Tree-of-Thought Approach in Prompt Engineering
- Lab 8: Controlling Verbosity and the Nova System
- Lab 9: Getting to Know watsonx Prompt Lab

Module 5: Capstone Project

Course Code	24HS04PR0101			
Category	Co-Curricular Activity			
Course Title	Sports-Yoga-Recreation			
Scheme & Credits	L	P	Credit	Semester
	0	2	1	I

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

Objectives of the Course:

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

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

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

Course Content:

Unit 1:

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

Unit 2:

- Yoga: Standing, Sitting, Prone & Supine positions.

- Suryanamaskar.
- Pranayama, Meditation and Relaxation Techniques.
- Recreational Games.
- Practice of Fundamental Skills of Basketball, Football, Carrom, etc.
- Health related Physical Fitness Test.

Assessment Pattern:

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

References:

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

Course Title: Liberal/Performing Arts

Course Code	CourseName	Sem	Hours /week	Credits	Continuous Evaluation Max. marks
24HS02PR0105-01	Fundamentals of Indian Classical Dance: Bharatnatayam	I	2	1	50
24HS02PR0105-02	Fundamentals of Indian Classical Dance: Kathak	I	2	1	50
24HS02PR0105-03	Introduction to Digital Photography	I	2	1	50
24HS02PR0105-04	Introduction to Basic Japanese Language	I	2	1	50
24HS02PR0105-05	Art of Theatre	I	2	1	50
24HS02PR0105-06	Introduction to French Language	I	2	1	50
24HS02PR0105-07	Introduction to Spanish Language	I	2	1	50
24HS02PR0105-08	Art of Painting	I	2	1	50
24HS02PR0105-09	Art of Drawing	I	2	1	50
24HS02PR0105-10	Nature Camp	I	2	1	50
24HS02PR0105-11	Developing Self-awareness	I	2	1	50
24HS02PR0105-12	Art of Poetry	I	2	1	50
24HS02PR0105-13	Creative and content writing	I	2	1	50
24HS02PR0105-14	Science of life through Bhagwad Gita	I	2	1	50
24HS02PR0105-16	Sanskrit Sambhashan- Spoken Sanskrit	I	2	1	50
24HS02PR0105-17	Kirtan Kala	I	2	1	50

Course Code	24HS02PR0105-01			
Category	Co-Curricular Activity			
Course Title	Fundamentals of Indian Classical Dance: Bharatnatayam			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

1.Orientation in Bharatnatayam
2.Tattu Adavu till 8, Naatta Adavu 4 Steps, PakkaAdavu 1 step, Metta Adavu 1 Step, Kuditta Metta Adavu4 Steps
3. Practice sessions
4.Tatta Kuditta Adavu(Metta),Tatta Kuditta Adavu(Metta)2Steps,TirmanamAdavu3Steps,KattuAdav-3 Steps,Kattu Adav-3 Steps
5. Practice sessions
6. Tiramanam (front) 3 Steps, Repeat of Tiramanam (Overhead) 3 Steps
7. Practice sessions
8.Final practice sessions and performances.

Recommended reading

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

Course Code	24HS02PR0105-02			
Category	Co-Curricular Activity			
Course Title	Fundamentals of Indian Classical Dance: Kathak			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

1.Orientation in Kathak. Correct posture of kathak, Basic Movements and exercise Stepping, Chakkar of 5count(Bhramari),
2.Practice sessions of practical 1
3.Hastaks, Hastaks and Steppings, Reciting as amyukta Mudrashloka, Hastak and steppings4.practicesessions ofpractical3
5.Todas and Asamyukta hasta mudra shlok, Vandana of Shlok, 2 Todas and Vandana, Ghante Ki Tihai,
6.practicesessions of practical5
7.21ChakkardarTodaand Ginnti Ki Tihai, 2 Todasand1 Chakkardar Toda,practice sessions
8.Final performances.

Recommended reading

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

Course Code	24HS02PR0105-03			
Category	Co-Curricular Activity			
Course Title	Introduction to Digital Photography			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

On successful completion of the course,students will be able to
1. Develop an understanding of the technical aspects and aesthetics of Photography.
2. Apply the rules of digital photography for creating photographs.
3. Develop skills to enhance photographs through postprocessing.
4. Create a portfolio of their photographs in selected genre.

Syllabus

1.Orientation in digital photography**:** Genres,camera handling and settings
2. Rules of Composition
3.Rules of Composition: practice sessions4.UnderstandingExposureandArtofPre-Visualization
5.Rules of Composition and Art of Pre-Visualization: practice sessions
6. Post Processing Photographs and Portfolio creation
7.Post Processing Photographs: practice sessions
8.Portfoliofinalizationandpresentationinselectedgenre.

Reference material

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

Course Code	24HS02PR0105-04			
Category	Co-Curricular Activity			
Course Title	Introduction to Japanese Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Recommended Reading

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

Course Code	24HS02PR0105-05			
Category	Co-Curricular Activity			
Course Title	Art of Theatre			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Reference Books

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

Course Code	24HS02PR0105-06			
Category	Co-Curricular Activity			
Course Title	Introduction to French Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

List of Practicals

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

Recommended Reading

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

Course Code	24HS02PR0105-07			
Category	Co-Curricular Activity			
Course Title	Introduction to Spanish Language			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

List of Practicals

1. Orientation about Spain, the language, and culture
2. Communication Skills 1: Vocabulary building for everyday conversations
3. Practice sessions
4. Reading and Writing Skills: Reading and writing simple text in Spanish
5. *— (Missing in original — assuming a numbering issue)*
6. Communication Skills 2: Listening comprehension
7. Practice sessions
8. Writing Skills: Write basic Spanish and practice

Recommended Reading

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

Course Code	24HS02PR0105-08			
Category	Co-Curricular Activity			
Course Title	Art of Painting			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Reference Material

1. *Drawing Made Easy* by Navneet Gala; 2015 edition
2. *Alla Prima II: Everything I Know About Painting – And More* by Richard Schmid with Katie Swatland
3. *Daily Painting: Paint Small and Often to Become a More Creative, Productive, and Successful Artist* by Carol Marine

Course Code	24HS02PR0105-09			
Category	Co-Curricular Activity			
Course Title	Art of Drawing			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Syllabus

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

---

Reference Material

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

Course Code	24HS02PR0105-10			
Category	Co-Curricular Activity			
Course Title	Nature Camp			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course Outcomes

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

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

Course Content

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

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

Course Code	24HS02PR0105-15			
Category	Co-Curricular Activity			
Course Title	Sanskrit Sambhashan- Spoken Sanskrit			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objectives:

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

Course outcome

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

CO1: Enhanced writing skills in Sanskrit
CO2: Enhanced speaking skills in Sanskrit
CO3: Enhanced listening skills in Sanskrit
CO4: Enhanced writing skills in Sanskrit

Syllabus:

संस्कृतसम्भाषणशिबिरस्य पाठ्यक्रमः

प्रथमं दिनम्

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

द्वितीयं दिनम्

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

तृतीयं दिनम्

- गीतम्।
- पुनस्स्मारणम्।
- क्रियापदानां बहुवचनरूपाणि।
  गच्छन्ति - गच्छामः - गच्छन्तु (Chart दर्शनीयम्)
  पिबन्ति - पिबामः - पिबन्तु।
  लिखन्ति - लिखामः - लिखन्तु।
  इत्यादिपरिवर्तनाभ्यासः कारणीयः।
- द्वितीयाविभक्तिः - स्फोरकपत्राणाम् उपयोगः।
  (वाक्यपत्राणि उपयुज्य वाक्यानि वाचनीयानि।)
- कृपया ददातु - वस्तूनि प्रदर्श्य।
  शिक्षकः एकैकं वस्तु प्रदर्शयति।
  उदा. – ग्रन्थः, घटी,.....
  छात्राः - कृपया ग्रन्थं ददातु, कृपया घटीं ददातु इत्यादि वदेयुः। (स्फोरकपत्रस्य उपयोगः)
- पुरतः, पृष्ठतः, वामतः, दक्षिणतः, उपरि, अधः।
  (चित्रं दर्शनीयम्)
- इतः, ततः, ...तः, गृहतः, कुतः?
  (स्फोरकपत्राणाम् उपयोगः)
  वाक्यपत्राणि उपयुज्य वाक्यानि वाचनीयानि।
- गीतम्।
- कथम्? सम्यक्।
- शीघ्रम् × मन्दम्। उच्चैः × शनैः।
- पठनार्थम्, किमर्थम्?
- सप्तककाराः - किम्, कुत्र, कति, कदा, कुतः, कथम्, किमर्थम् (Chart प्रदर्शनीयम्)।
  एकैकम् उपयुज्य परस्परं प्रश्नाः।
- अपि।
- अस्तु।
- अहं न जानामि। - कानिचन वाक्यानि।
- भूतकालीनक्रियापदानां पाठनम्।
  गतवान् - पठितवान् - लिखितवान्।
  गतवती - पठितवती - लिखितवती।
- क्रियापदकोष्ठकस्य प्रथमपृष्ठस्य अभ्यासः।
- द्वितीयपृष्ठस्य सर्वाणि क्रियापदानि उपयुज्य छात्राः वर्तमानकाले वाक्यानि वदन्ति। (ए.व - ब.व.)
- विशिष्टक्रियापदानाम् अभ्यासः -
  करोमि - कुर्मः। करोति - कुर्वन्ति।
  ददामि - दद्मः। ददाति - ददति।
  शृणोमि – शृणुमः। शृणोति - शृण्वन्ति।
  जानामि – जानीमः। जानाति – जानन्ति।
- सम्बोधनम् - भोः !, श्रीमन् !, मान्ये !, भगिनि!, मित्र !, ...महोदय!, राम !, सीते ! इत्यादि।
- सङ्ख्या- 1-100।
- समयः - 1.00, 2.00, 3.00, 4.00।
- सम्भाषणप्रदर्शनम् (मित्रसंलापः)।
- कथा।
- वाक्यत्रयम् एकैकोऽपि छात्रः वदेत्।
- सूचना।
- ऐक्यमत्यः।

चतुर्थं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- च
- अतः
- एव
- इति
- अस्मि
- यदि -तर्हि
- यथा - तथा
- तः - पर्यन्तम् (वाक्यपत्रस्य उपयोगेन वाक्यानि वाचनीयानि।)
- अद्य आरभ्य
- कृते (वाक्यपत्रस्य उपयोगः करणीयः)
- क्तवतुप्रत्ययान्तानाम् अभ्यासः
  गतवान् - पठितवान् - लिखितवान् (ए.व. पुंलिङ्गे)।
  गतवती - पठितवती - लिखितवती (ए.व. स्त्रीलिङ्गे)।
  गतवन्तः - पठितवन्तः - लिखितवन्तः (ब.व. पुंलिङ्गे)।
  गतवत्यः - पठितवत्यः - लिखितवत्यः (ब.व. स्त्रीलिङ्गे)।
- सः गतवान् - सा गतवती - लिङ्गपरिवर्तनाभ्यासः।
- अहं गतवान् - अहं गतवती - लिङ्गपरिवर्तनाभ्यासः।
- क्रियापदानां कालपरिवर्तनाभ्यासः।
  यथा - गच्छति - गतवान्, गतवती।
- गीतम्।
- विशेषपाठनम् - आसीत्, आसन्, आसम्।
- एकः, एका, एकम् - लिङ्गभेदः ज्ञापनीयः।
  (स्फोरकपत्रस्य उपयोगः)

- भोजनसम्बन्धिशब्दाः यथा - सूपः, शाकम्, इत्यदयः।
- सङ्ख्या।
- समयः।
- ॐ – सङ्ख्याक्रीडा।
- कथा।
- सम्भाषणप्रदर्शनम्।
- चत्वारि वाक्यानि।
- सूचना।
- ऐक्यमत्यः।

पञ्चमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- वाहनानां नामानि।
- तृतीयाविभक्तिः - दण्डेन, मापिकया, लेखन्या, पुष्पेण।
  (वाक्यपत्रस्य आधारेण वाक्यानि वाचनीयानि।)
- सह, विना।
- अद्यतन, ह्यस्तन, श्वस्तन, पूर्वतन, इदानीन्तन
- भविष्यत्कालीनक्रियापदानां पाठनम्।
  गमिष्यति, पठिष्यति, लेखिष्यति।(कोष्ठकस्य साहाय्येन)
- गत,आगामि।
- गीतम्।
- स्म।
- अभवत्।
- क्त्वाप्रयोगः (कोष्ठकस्य साहाय्येन)।
- यदा – तदा।
- बन्धुवाचकशब्दाः।
- वेशभूषणानां नामानि।
- वर्णाः।
- रुचयः।
- क्रीडा - एकश्वासेन सङ्ख्याकथनम्।
- कथा।
- पञ्च वाक्यानि।
- सूचना।
- ऐक्यमत्यः।

षष्ठं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- नूतनम् x पुरातनम्,
- बहु x किञ्चित्,
- दीर्घः x ह्रस्वः।
- उन्नतः x वामनः।
- स्थूलः x कृशः।
- एतादृश, तादृश, कीदृश?
- तुमुन् (कोष्ठकस्य साहाय्येन)।
- किन्तु।
- निश्चयेन।
- बहुशः / प्रायशः।
- किल / खलु।
- शक्नोति।
- गीतम्।
- विशेषणविशेष्यभावस्य अभ्यासः।(प्रथमाविभक्तौ)
  सः उत्तमः बालकः।
  सा उत्तमा बालिका।
  तत् उत्तमं पुस्तकम्।
- इव। विनोदकणिका।(गतवान् 'इव' अभिनयं कृतवान्!)
- अपेक्षया।
- पशूनां नामानि।
- अवयवानां नामानि।
- वाक्यविस्तारणाभ्यासः।
  (सः मम पुस्तकं प्रातःकाले पञ्चवादने पठितवान्।)
- इतः पूर्वम् - इतः परम्।
- 'रामकृष्ण' सङ्ख्याक्रीडा।
- कथा।
- षट् वाक्यानि।
- सूचना।
- ऐक्यमत्यः।

सप्तमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- क्त्वा - तुमुन् - परिवर्तनाभ्यासः।
- बहिः x अन्तः।
- रिक्तम् x पूर्णम्।
- इतोऽपि।
- इत्युक्ते।
- अन्ते।
- चेत् - नो चेत्।

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

अष्टमं दिनम्

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

नवमं दिनम्

- गीतम्।
- पुनःस्मारणम्।
- चित्।
- ....द्वयम्।
- सङ्ख्यासु लिङ्गभेदः।
  एकः - एका - एकम्
  द्वयम् - द्वयम् - द्वयम्
  त्रयः - तिस्रः - त्रीणि
  चत्वारः - चतस्रः - चत्वारि
- शिश्रकः - अहं वैद्यः - मम नाम सुरेशः
  (छात्राः तमुद्दिश्य प्रश्नान् पृच्छेयुः।)
- ..... अर्थम् (समाजार्थम्, संस्कृतकार्यार्थम्...)।
- गीतम्।
- तव्यत् – अनीयर्।
- अनन्त्यकथारचना।
- सङ्ख्यान्वेषणम् (क्रीडा)।
- छात्रैः सह प्रश्नोत्तरम्।
- समाजनिधिविषये पुनःस्मारणम्।
- ऐक्यमन्त्रः।

दशमं दिनम्

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

Course Code	24HS02PR0105-16			
Category	Co-Curricular Activity			
Course Title	Kirtan Kala			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	I

Course objectives:

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

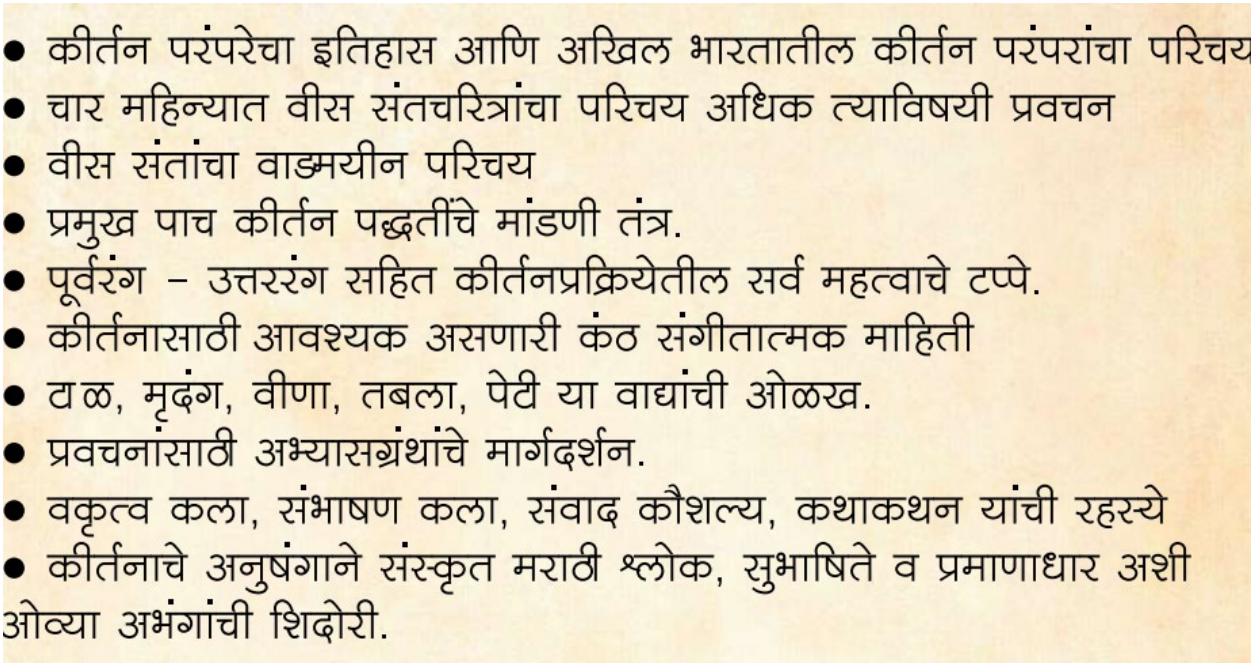
Course outcome

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

CO1: Learn from the inspiring spiritual journey of the saints and the history of Kirtan tradition

CO2: Learn about the musical instruments used in the art of Kirtan

CO3: Develop communication skills



Course Code	24EE01TH0106/24EE01PR0106			
Category	Vocational and Skill Enhancement Course			
Course Title	Creativity Innovation and Design Thinking			
Scheme & Credits	L	P	Credit	Semester
	1	2	2	I

Course Outcomes:

1. Be familiar with processes and methods of creative problem solving
2. Enhance their creative and innovative thinking skills
3. Practice thinking creatively and innovative design and development

---

Syllabus:

Module I: Introduction
Making a case for creativity, Creative thinking as a skill, Valuing diversity in thinking: Thinking preferences, Creativity styles, Creativity in problem solving

Module II: Pattern Breaking
Thinking differently, Lateral thinking, Mind stimulation: Games, brain-twisters and puzzles, Idea-collection processes, Brainstorming/Brain-writing, The SCAMPER methods, Metaphoric thinking, Outrageous thinking, Mapping thoughts, Other (new approaches)

Module III: Using Math and Science
Systematic logical thinking, Using math concepts, Eight-Dimensional (8D) Approach to Ideation: Uniqueness, Dimensionality, Directionality, Consolidation, Segmentation, Modification, Similarity, Experimentation

Module IV: Systematic Inventive Thinking
Systematic inventive thinking: The TRIZ methodology, Decision and Evaluation: Focused thinking framework, Six thinking hats, Ethical considerations

Module V: Design for Innovation
Introduction to design for interaction, Nine lessons for innovation, Difference in creativity and innovation, Building blocks for innovation

Module VI: Intellectual Property
Introduction to intellectual property: Patents, Copyrights, Trademarks, Trade Secret, Unfair Competition

---

Textbook and Reference Books:

1. *Creative Problem Solving for Managers* – Tony Proctor, Routledge Taylor & Francis Group
2. *101 Activities for Teaching Creativity and Problem Solving* – By Arthur B. Van Gundy, Pfeiffer
3. H. S. Fogler and S. E. LeBlanc, *Strategies for Creative Problem Solving*, Prentice Hall
4. E. Lumsdaine and M. Lumsdaine, *Creative Problem Solving*, McGraw-Hill
5. J. Goldenberg and D. Mazursky, *Creativity in Product Innovation*, Cambridge University Press, 2002

Course Code	24EE01TH0201/24EE01PR0201			
Category	Programme Core Course			
Course Title	CMOS Digital Circuit Design			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

1. Acquire knowledge about various NMOS, PMOS and CMOS digital circuits and interconnects
2. Implement digital logic structure of various types
3. Estimate various performance metrics for digital circuits.
4. Analyse memory elements.
5. Analyse performance of moderately sized CMOS circuits by using modern tools to verify the functionality, timing, power and parasitics using schematic and/or layout simulation for a given technology.

Syllabus

Module I: (8 Hours)

Overview of VLSI Design Methodology, Design Flow & hierarchy, Introduction to MOS Transistors, Threshold voltage, body effect, MOS device design equations, second order effects, MOS Models-Level-1, Level-2, Level-3.

Module II: (8 Hours)

Static Load MOS Inverters, CMOS Inverter: The Static Behavior, Switching threshold, Noise Margins, The Dynamic Behavior, Power, Energy, and Energy-Delay, the Tri State Inverter, Transmission Gate. CMOS fabrication process flow, N-well, P-well, Twin-tub process flow, Silicon on insulator, Latch-up, Layout design rules (DRC).

Module III: (7 Hours)

Circuit Characterization and Performance Estimation: Introduction, Resistance Estimation Capacitance Estimation, CMOS gate transistor sizing, Driving Large capacitive loads, Scaling of MOS transistors.

Module IV: (9 Hours)

Designing combinational logic gates in CMOS: Complementary CMOS, Ratioed Logic, Pass- Transistor Logic, Dynamic CMOS Design, Dynamic Logic: Basic Principles, Issues in Dynamic Design, Cascading of Dynamic Gates, Domino Logic.

Module V: (7 Hours)

Sequential logic design: Timing Metrics for Sequential Circuits, Classification of Memory Elements, Static Latches and Registers, Dynamic Latches and Registers

Module VI: (6 Hours)
Clocking Strategies, CMOS Sub-system design: SRAM, DRAM.

Text Book

1.Digital Integrated Circuits: A Design Perspective: J. Rabaey, 2nd edition PHI

Reference Books

1. CMOS VLSI Design: A circuits and systems perspective: N. Weste and K. Eshranghian, 2nd edition, PHI

2. CMOS Digital Integrated Circuits Analysis & Design: S M Kang, Yusuf Lablebici, 3rd edition TMH3.V L S I Design Technique for Analog and Digital Circuit: Randel Geiger, P Allen, N Strader, 2nd edition TMH

3. Introduction to VLSI System: Carver Mead, Lynn Conway, 1st edition Addison- Wesley

4. MOS Integrated Circuits- Theory, Fabrication, Design and System Applications of MOS LSI: William M. Penny, Lillian Lau, Van Nostrand Reihold Company. 1st edition

5. Basic VLSI Systems and Circuits: Dougles Pucknell and K. Eshraghian 3rd edition, PHI

List of Experiments:

1)Evaluate the output characteristics for 1um (long channel) and 50nm (short channel) technology N-channel and P-channel MOSFET by plotting the output characteristics.

2)Using SPICE, plot the transfer characteristics for the inverter seen in Fig.2 in both the long- and short-channel CMOS technology. From the plot, determine V_M, V_{IL}, V_{IH}, V_{0H}, and V_{0L}. Calculate noise margin in both the cases. Use VDD=5V for long channel devices and VDD=1V for short channel devices. Comment on dynamic power consumption.

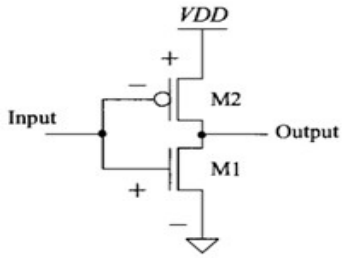


Fig.2

3)Investigate using SPICE and the long-channel CMOS process parameters supplied to you, the transfer curves for the CMOS inverter with trans-conductance ratios βn/βp 3, 1, and 1/3. Explain what changing the ratio does to the transfer characteristics. Estimate and simulate the intrinsic propagation delays of inverter if short channel devices are used (PMOS: NMOS=2:1).

4)Draw the layout of inverter designed for equal rise and fall time and extract the layout and simulate the netlist in SPICE.

5)Design a buffer (Chain of Inverter) to drive the load capacitance with a minimum delay. You need to start the design from the minimum sized inverter. Load that needs to be driven is of 10 pF.

6)Implement D F/F. Define and estimate setup time for your design through SPICE simulation.

7)To design and simulate SRAM cell with sense amplifier circuit.

8)Design a Ring-Oscillator in 50nm CMOS technology. Study the effect of process variations on oscillation frequency.

Course Code	24HS03TH0217			
Category	Basic Science Course			
Course Title	Probability and Statistics			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	II

Course Outcomes

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

1. Grasp the meaning of discrete and continuous random variables, probability distribution. Interpret the meaning of probabilities derived from distributions. This involves understanding what the calculated probabilities represent in practical terms and drawing conclusions from the results.
2. To analyze and interpret stochastic models, including calculating probabilities, transition probabilities, and steady-state probabilities within stochastic systems.
3. Grasp the fundamental concepts of curve fitting like regression techniques, model selection, and the use of different types of curves or functions to approximate data.
4. Understand the fundamental concept of hypothesis testing, including the null hypothesis (H0) and alternative hypothesis (H1), significance levels, p-values, and the basic logic behind hypothesis testing.
5. To apply MLE to various statistical models, such as linear regression, exponential distribution, etc. They should understand how to formulate likelihood functions and derive estimators for unknown parameters.

Syllabus

Module 1(8 hours)

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

Module 2:(8 Lectures)

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

Module 3: (8 hours)

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

Module 4: (8 Lectures)

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

Module 5: (7 Lectures):

Testing of Hypothesis for difference of mean and proportion, Test for ratio of variances - Chi-square test

for goodness of fit and independence of attributes, maximum likelihood estimation

Text Books:

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

Reference Books:

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

Course Code	24EE01TH0202/24EE01PR0202			
Category	Programme Core Course			
Course Title	Digital System Design			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

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

Syllabus

Module I (6 Hrs)

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

Module II (8 Hrs)

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

Module III (6 Hrs)

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

Module IV (6 Hrs)

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

Module V (6 Hrs)

Design of Data Path and Control unit with Case Studies.

Module VI (6 Hrs)

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

Text Book

1. Verilog HDL: A Guide to Digital Design and Synthesis; Samir Palnitkar, Prentice Hall PTR; 2nd Edition
2. Fundamentals of Digital Logic with Verilog; Stephen Brown and ZvonkoVranesic; McGraw Hill, 2nd Edition

Reference Books

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

List of Experiments

1. Design the following combinational circuits using Verilog HDL. Write a functional simulation test bench to test these designs. Generate all combinations of test inputs with a delay of 5 ns between the set of each test input. Observe the output in the console window.

a. Full adder using operators b. Full adder using two Half adders c . 4:1 Mux using 2:1 Mux
Synthesize the design and count the number of LUT. Plan a pin constraint for any one design for implementation on FPGA.

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

3. Model the 4 bit counter with following specifications:

a. Reset synchronous to clock positive edge.
b. Counter has a 4 bit load terminal to count Up or down from this user loaded value.
Write a test bench to test this design and implement it on FPGA.

4. Design a 4 bit sequence detector using verilog HDL. Write a test bench to test this design.
5. Create a Verilog RAM module with 8 bit wide data and 16 locations, initialize it with data, and perform a read operation to verify data retrieval. Find the average of the retrieved data in the console window.

Course Code	24EE01TH0203/24EE01PR0203			
Category	Engineering Science Course			
Course Title	Elements of IoT			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

Module 1: Introduction to Internet of Everything, IoT Reference Model, Different IoT models, Elements in IoT Infrastructure

Module 2: IoT Infrastructure Elements and their roles at Different Layer in IoT Reference Model, Devices/ Function of elements in IoT Sensors, Controllers, Network, Cloud, User Applications and Data Analytics

Module 3: Perception Layer, Network Layer, Application Layer Architecture in IoT system

Module 4: Resources used at Perception Layer, Network Layer, Application Layer

Module 5: Use cases of IoT Systems builds across SAM IoT/ Arduino Nano IOT/ ESP32 / Node MCU/ PI-PICO H/W variants etc,

Text Book:

1. Internet of Things Principles and Paradigms, Rajkumar Buyya Amir Vahid Dastjerdi, Morgan Kaufman, Elsevier 2016 1st Edition

Reference Books:

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

List of Experiments

The hands-on lab will be based on the following

1. Understand the IoT reference Model implementation on various use cases of IoT System in applications specific domains.

2. Identification of elements used in various use cases

3. Analyse the role and functions of different IoT elements used in different use cases

4. Proposing a solution based on the IoT reference frame work

5. Evaluating the engineering feasibility of the IoT solution on the basis of
    a. Hardware platform/ resource usage
    b. Communication mode used
    c. Uptime requirements
    d. Scale and volume of data
    e. Security and Maturity

Course Code	24EE01TH0204/24EE01PR0204			
Category	Programme Core Course			
Course Title	Object Oriented Programming			
Scheme & Credits	L	P	Credits	Semester
	3	2	4	II

Course Outcomes

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

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

Syllabus

Module 1
Features of Object-Oriented Programming languages: Abstraction, Encapsulation, Inheritance, Polymorphism, and Late Binding. Concept of a class, Access control of members of a class, Instantiating a class, Constructor and method overloading and overriding.

Module 2
Concept of inheritance, methods of derivation, use of `super` keyword and `final` keyword in inheritance, runtime polymorphism, abstract classes and methods, Interface, implementation of interface.

Module 3
Creating packages, importing packages, static and non-static members, Lambda Expressions Introduction, Block, Passing Lambda expression as Argument.

Module 4
Exceptions: Types of exceptions, use of try-catch block, handling multiple exceptions, using finally, throw and throws clause, user-defined exceptions, file handling in Java, Serialization.

Module 5
Generics: Generic class with two type parameters, bounded generics. Collection classes: ArrayList, LinkedList, HashSet, TreeSet, HashMap.

Module 6
Multithreading: Java Thread models, creating thread using runnable interface and extending Thread, thread priorities,

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

Text Books

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

List of Experiments

1. Implement the Classes and Objects in Java.
2. Implement a program in Java with Constructors and destructors. Also implement the concept of overloading.
3. Demonstrate use of Inheritance.
4. Implement a concept of Interface in Java.
5. Demonstrate use of MultiThread.
6. Implement Packages and import that package in program.
7. Demonstrate use of Exception handling mechanism.
8. Apply concept of generics class and method.
9. Demonstrate collection framework and perform some basic operations on the ArrayList and HashSet.
10. Apply file handling concepts in Java.

Course Code	24EE01PR0205			
Category	Vocational and Skill Enhancement Course			
Course Title	Computer Workshop- II			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	II

Course Outcomes

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

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

Practicals based on:

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

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

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

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

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

Remote access and managing processes through remote login

Text books

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

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

Course Code	24HS02TH0201/24HS02PR0201			
Category	Ability Enhancement Course			
Course Title	English for Professional Communication			
Scheme & Credits	L	P	Credits	Semester
	2	2	3	II

Course Outcomes

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

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

Syllabus

Module 1: Vocabulary Building

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

Module 2: Listening and Reading Comprehension

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

Module 3: Functional Grammar and Usage

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

Module 4: Writing Skills

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

Module 5: Writing Practices

- Art of Condensation: Précis, Summary, and Note Making
- Correspondence writing techniques and etiquettes – academic writing, Essay Writing

---

Reference Books

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

---

Course Outcomes

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

1. Apply effective listening and speaking skills in professional and everyday conversations.
2. Demonstrate the techniques of effective Presentation Skills.
3. Evaluate and apply the effective strategies for Group Discussions.
4. Analyse and apply the effective strategies for Personal Interviews.
5. Implement essential language skills – listening, speaking, reading, and writing.

List of Practicals

Computer Assisted and Activity Based Language Learning

1. Everyday Situations: Conversations and Dialogues – Speaking skills
2. Pronunciation, Intonation, Stress, and Rhythm
3. Everyday Situations: Conversations and Dialogues – Listening Skills

Activity Based Language Learning

4. Presentation Skills: Orientation & Mock Session
5. Presentation Skills: Practice
6. Group Discussions: Orientation & Mock Session
7. Group Discussions: Practice
8. Personal Interviews: Orientation & Mock Session
9. Personal Interviews: Practice

Course Code	24HS02TH0203			
Category	Indian Knowledge System			
Course Title	Foundational Literature of Indian Civilization			
Scheme & Credits	L	P	Credits	Semester
	1	0	1	II

Course Outcomes

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

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

Syllabus

Module 1: Overview of Indian Knowledge System
Importance of ancient knowledge, defining IKS, IKS classification framework, Historicity of IKS, Some unique aspects of IKS.

Module 2: The Vedic Corpus
Introduction of Vedas, four Vedas, divisions of four Vedas, six Vedangas, Distinct features of Vedic life.

Module 3: Indian Philosophical Systems
Development and unique features, Vedic schools of philosophy, Samkhya and Yoga School of philosophy, Nyaya and Vaisheshika school of philosophy, Purva-mimamsa and Vedanta schools of Philosophy, Non-vedic philosophies: Jainism, Buddhism, and other approaches.

Module 4: Indian Wisdom Through Ages
Panchatantras, Puranas: contents and issues of interests, Itihasa: uniqueness of the two epics (Ramayana and Mahabharata), Key issues and messages from Ramayana, Mahabharata – a source of worldly wisdom; Indian ancient Sanskrit literature: Kalidas, Vishakadutta, Bhavbhuti, Shudraka — any one text as decided by the course teacher.

Reference Material

1. B. Mahadevan, Vinayak Rajat Bhar, Nagendra Pavana R. N., *Introduction to Indian Knowledge System: Concepts and Applications*, PHI, 2022

2. S. C. Chatterjee and D. M. Datta, *An Introduction to Indian Philosophy*, University of Calcutta, 1984

Course Code	**24EE01TH0301**			
Category	**Programme Core Course**			
Course Title	**Data Structures and Algorithms**			
Scheme & Credits	**L**	**P**	**Credits**	**Semester**
	3	**0**	**3**	**III**

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

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

Syllabus:

Module 1: (06 Hrs)
Data Structures and Algorithm Basics:
Introduction: basic terminologies, elementary data organizations, data structure operations; abstract data types (ADT) and their characteristics. Algorithms: definition, characteristics, analysis of an algorithm, asymptotic notations, time and space tradeoffs. Array ADT: definition, operations and representations – row-major and column-major.

Module 2: (06 Hrs)
Stacks and Queues:
Stack ADT: allowable operations, algorithms and their complexity analysis, applications of stacks – expression conversion and evaluation (algorithmic analysis), multiple stacks. Queue ADT: allowable operations, algorithms and their complexity analysis for simple queue and circular queue, introduction to double-ended queues and priority queues.

Module 3: (06 Hrs)
Linked Lists:
Singly Linked Lists: representation in memory, algorithms of several operations: traversing, searching, insertion, deletion, reversal, ordering, etc. Linked representation of stacks and queues, header node linked lists. Doubly and Circular Linked Lists: operations and algorithmic analysis.

Module 4: (06 Hrs)

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

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

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

Text Book:

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

Reference Books:

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

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

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

Course Code	24EE01PR0301			
Category	Programme Core Course			
Course Title	Data Structures and Algorithms			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	III

List of Experiments:

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

Course Code	24EE01TH0302			
Category	Programme Core Course			
Course Title	Computer Architecture			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes:

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

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

Syllabus:

Module-I:

Introduction to computer system and its sub modules, Introduction to RISC and CISC paradigm, Performance Equation, Common Principles of Computer organization: Amdahl's Law, Principle of Locality.

Module-II:

Processor organization, instruction set (RISC-V), instruction formats, Representing Instructions in the Computer, Translating and Starting a Program, Arithmetic for Computers: Addition and Subtraction, Multiplication, Division, IEEE 754 floating point format.

Module-III:

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

Module-IV:

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

Pipelining Hazards.

Module-V:
Memory organization, concepts of semiconductor memory, memory management, concept of cache and associative memories, virtual memory.

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

Text Books:

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

Reference Books:

1. Computer Architecture and Organization; J. P. Hayes; Third Edition (Fifth Reprint), McGraw Hill, 2012.

2. Computer Architecture And Parallel Processing; Kai Hawang, Faye A. Briggs, McGraw Hill, 2012

Course Code	24EE01PR0302			
Category	Programme Core Course			
Course Title	Computer Architecture Lab			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	III

List of Experiments:

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

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

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

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

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

Course Code	24EE01TH0303			
Category	Basic Science Course			
Course Title	Discrete Mathematics			
Scheme &Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes:

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

1. Comprehend modular arithmetic, recurrence relations, generating functions to solve problems.
2. Understand a given problem of graph network and solve with techniques of graph theory.
3. Realize the lattice as algebraic structure and use it for pattern recognition in cryptography.
4. Apply groups and fields in coding theory.

Syllabus:

Module I (7Hrs) Modular Arithmetic:

Modular Arithmetic, Euclid's Algorithm, Primes, Fermat's Algorithm, Euler's Theorem, Linear Congruences, Chinese Remainder Theorem, Application to Cryptography.

Module II (8Hrs) Combinatorics:

Addition and multiplication rules in combinatories, Linear and circular permutation, Combination, Inclusion and Exclusion Principle, recurrence relations, generating function, examples using ordinary power series and exponential generating functions.

Module III (7Hrs) Graph Theory:

Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub-Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Coloring, Coloring maps and Planar Graphs, Perfect Graph.

Module IV (7Hrs) Lattice theory:

Lattices as partially ordered set, Definitions and Examples, some properties of Lattices, Lattices as algebraic system, sub lattices, direct product, homomorphism, some special Lattices.

Module V (7Hrs) Groups and Fields:

Group definitions and examples, cyclic group, permutation groups, subgroups and homomorphism, co-sets, Lagrange's theorem, Finite field.

Text Books:

1. Seymour Lipschutz and Marc Lars Lipson Adapted by Varsha H. Patil, Discrete Mathematics, Revised 3rd edition Schaum's outlines, Tata McGraw-hill Publication
2. J. P.Tremblay and R. Manohar; Discrete Mathematical Structures with Applications to Computer Science; TataMcGraw-hill Publication.

Reference Books:

1. 1.Kenneth H. Rosen, Discrete Mathematics and its Applications, 8th edition Tata McGraw – Hill
2. Susanna S. Epp, Discrete Mathematics with Applications, 4th edition, Wadsworth Publishing Co.Inc.
3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw –Hill.

Course Code	24EE01TH0304			
Category	Programme Core Course			
Course Title	Digital Signal Processing			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

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

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

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

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

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

Syllabus:

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

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

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

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

Module V (5 Hrs):
Design of IIR filter: Impulse invariance transformation, Bilinear Transformation, Design of Butterworth and Chebyshev filters, structures for IIR systems.

Module VI (5 Hrs):
DSP hardware, Finite word length effects, Multirate signal processing: Quantization by truncation and Rounding, Quantization of Input data and filter coefficients, Digital Signal Processing applications, introduction to DSP processors, multirate signal processing.

Text Book:

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

Reference Book:

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

Course Code	**24EE01PR0304**			
Category	**Programme Core Course**			
Course Title	**Digital Signal Processing Lab**			
Scheme & Credits	**L**	**P**	**Credits**	**Semester**
	0	**2**	**1**	**III**

Course Outcomes:

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

1. Describe discrete-time signals in different forms and analyze the behavior of Linear Time-Invariant (LTI) systems in the frequency domain.
2. Apply Z-transform techniques to process signals in various discrete-time systems, and interpret their system behavior.
3. Design and implement Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, and evaluate their performance to ensure they meet expected system specifications.
4. Analyze finite word-length effects due to rounding and truncation, examine the impact of multi-rate signal processing, and understand the fundamentals of DSP processors.

Syllabus:

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

List of Experiments

1. Generate and visualize basic discrete-time signals (unit step, impulse, ramp, sinusoidal, exponential).
2. Perform linear convolution and circular convolution of signals and compare the results.
3. Compute and analyze frequency response of a given LTI system using DFT/FFT.
4. Compute Z-transform and its inverse for given discrete-time signals.
5. Analyze the stability of a system by plotting poles and zeros using the Z-plane representation.
6. Design and implement FIR filters using the windowing method (Hamming, Hanning, Blackman) and analyze their frequency response.
7. Design and implement IIR filters using the bilinear transformation and impulse invariance method.
8. Compare the performance of FIR and IIR filters in terms of magnitude response, phase response, and stability.
9. Analyze the effects of quantization, truncation, and rounding on digital signal processing.
10. Implement downsampling (decimation) and upsampling (interpolation) techniques and observe their impact on signal properties.
11. Implement basic DSP operations (filtering, FFT, convolution) using a DSP processor (like TMS320C67xx, ARM Cortex DSP).

Summary of Tool :

Experiment Category	Software Tools	Hardware Platforms
Signal Generation & Analysis	MATLAB, Python	--
Z-Transform Analysis	MATLAB, Python	--
Filter Design (FIR/IIR)	MATLAB, GNU Radio	FPGA, DSP Processors
Multi-Rate Signal Processing	MATLAB, Python, GNU Radio	DSP Kit, FPGA
Real-Time DSP Implementation	Code Composer Studio, Xilinx Vivado	TMS320C67xx, ARM Cortex

Course Code	24EE01TH0305-01			
Category	Multidisciplinary Minor -1			
Course Title	Basics of Chip Design using Verilog HDL			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes:

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

1. Realize the digital systems using Verilog HDL
2. Apply the testing strategies using HDL
3. Write a synthesizable HDL code for EDA tools
4. Analyze the timing issues in digital systems
5. Implement the digital systems on FPGA platforms.

Syllabus:

Module I (6 Hrs):

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

Module II (8 Hrs):

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

Module III (6 Hrs):

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

Module IV (6 Hrs):

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

Module V (6 Hrs):

Design of Data Path and Control unit with Case Studies.

Module VI (6 Hrs):

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

Text Book:

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

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

Reference Books:

1. Digital Systems Design Using Verilog; Charles Roth, Lizy K. John, ByeongKil Lee; Cengage Learning 2nd Edition

2. A Verilog HDL Primer: J Bhaskar; Star Galaxy Publishing; 2nd Edition.

Course Code	24EE01TH0305-02			
Category	Multidisciplinary Minor -2			
Course Title	Introduction to IoT system Design			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	III

Course Outcomes:

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

Syllabus:

Module 1:

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

Module 2:

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

Module 3:

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

Module 4:

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

Module 5:

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

Text Book:

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

Reference Books:

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

Syllabus of IV Semester

Course Code	24EE01TH0401			
Category	Programme Core Course			
Course Title	Embedded System Design			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

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

1. Understand the architecture and organization of Cortex microcontroller and its programming.
2. Acquire the knowledge, techniques and skill to integrate microcontroller hardware and software.
3. Analyse the concept of real time operating system architecture.
4. Apply microcontroller-based Embedded system knowledge to real world application.

Syllabus:

Module I: **(05 Hrs)**
Introduction to embedded System, RISC Principles, ARM Processor Families, AMBA Bus Architecture.

Module II: (06 Hrs)
The Cortex - M processor: Simplified view block diagram, programming model – Registers, Operation modes, Exceptions and Interrupts, Reset Sequence, Instruction Set, Pipeline, Bus, Priority, Vector Tables, Interrupt Inputs and Pending behavior, Fault Exceptions, Supervisor and Pendable Service Call, Nested Vectored Interrupt Controller.

Module III: (06 Hrs)
Introduction to the Cortex microcontroller software interface standard (CMSIS), Interfacing of GPIOs, Timers, ADC, PWM.

Module IV: (06 Hrs)
Communication Protocols: I2C, SPI, UART, MODBUS, USB and its Interfacing with Cortex - M Microcontrollers.

Module V: (06 Hrs)
RTOS Concepts-Critical section, Shared Resources, Context Switching, Pre-emptive and non-pre-emptive Schedulers, Priority Inversion, Mutual exclusion, Synchronization, Inter task communication mechanisms.

Module VI: **(06 Hrs)**
Structure of µCOS-II: Introduction to µCOS-II-, kernel structure, Task States, Inter task communication, Task Scheduling, Task Synchronization, Critical section, Shared Resources, Context

Switching, Priority Inversion, Mutual exclusion. Introduction to embedded Linux.

Text books:

1.The Definitive Guide to the ARM Cortex-M0: Joseph Yiu, Elsevier, (1/E) 2011.

Reference Books:

1. Freescale ARM Cortex-M Embedded Programming, Mazidi and Naimi ARM
2. An embedded software primer: David E Simon, Pearson education Asia, 2001
3. Micro C/OS II The Real Time Kernel: Jean J. Labrosse, CMPBooks,(2/E) 2002
4. Embedded Linux Primer: christopher Hallinan, Pearson (1/E) 2007

Course Code	24EE01PR0401			
Category	Programme Core Course			
Course Title	Embedded System Design Lab			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	IV

Experiment List:

PART-A:

Conduct the following experiments by writing Assembly Language Program (ALP) using ARM Cortex M0 Registers using an evaluation board/simulator and the required software tool.

1Write an ALP to multiply two 16-bit binary numbers.

2 Write an ALP to find the sum of first 10 integer numbers

3. Write an ALP to find factorial of a number.

4. Write an ALP to add an array of 16-bit numbers and store the 32-bit result in internal RAM

5. Write an ALP to add two 64-bit numbers.

6. Write an ALP to find the square of a number (1 to 10) using look-up table.

7. Write an ALP to find the largest/smallest number in an array of 32 numbers.

8. Write an ALP to arrange a series of 32-bit numbers in ascending/descending order.

9. Write an ALP to count the number of ones and zeros in two consecutive memory locations. 10. Write an ALP to Scan a series of 32-bit numbers to find how many are negative.

PART-B:

Conduct the following experiments on an ARM CORTEX M0 evaluation board using evaluation version of Embedded 'C' & Keil Uvision-5 tool/compiler.

1. Display “Hello World” message using Internal UART.

2. Interface and Control a DC Motor.

3. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.

4. Determine Digital output for a given Analog input using Internal ADC of ARM controller.

5. Interface a DAC and generate Triangular and Square waveforms.

6. Interface a 4x4 keyboard and display the key code on an LCD.

7. Using the Internal PWM module of ARM controller generate PWM and vary its duty cycle.

8. Demonstrate the use of an external interrupt to toggle an LED On/Off.

9. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.

10. Interface a simple Switch and display its status through Relay, Buzzer and LED.

Course Code	24EE01TH0402			
Category	Programme Core Course			
Course Title	Operating System			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:

After completion of the course student will be able to:

1. Understand the fundamental concepts and functions of operating systems.
2. Analyze process management, scheduling algorithms, and thread synchronization techniques.
3. Evaluate memory management strategies, including virtual memory and memory protection.
4. Implement file systems, disk scheduling algorithms, and device management techniques.
5. Analyze the principles of virtualization, types of virtual machines, their implementations, and the role of virtualization in modern operating systems, including mobile OS like iOS and Android

Syllabus:

Module I: Introduction to Operating Systems: Overview of operating systems, Role, and functions of operating systems, Types of operating systems, Historical perspective, Operating system structure and components.

Module II: Process Management, Scheduling Algorithms, Process Synchronization, Threads, and Deadlocks: Process concept and management, Process states and transitions, Process scheduling algorithms, CPU scheduling, Process synchronization and concurrency, Inter-process communication, Deadlock detection and prevention, Resource allocation and management, Multiprogramming and multitasking.

Module III: Memory Management: Memory hierarchy, Memory allocation strategies, Virtual memory concept, Paging and segmentation, Memory protection and addressing, Memory management unit (MMU).

Module IV: File Systems and I/O Management: File system organization and structure, File system implementation techniques, File system operations, Disk scheduling algorithms, Device management and drivers, Input/output operations and buffering.

Module V: Virtual Machines – History, Benefits and Features, Building Blocks, Types of Virtual Machines and their Implementations, Virtualization and Operating-System Components; Mobile OS

– iOS and Android.

Text Books:

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. Ramaz Elmasri, A. Gil Carrick, David Levine, “ Operating Systems – A Spiral Approach”, Tata McGraw Hill Edition, 2010.
2. William Stallings, “Operating Systems: Internals and Design Principles”, 7th Edition, Prentice Hall, 2018.
3. Achyut S. Godbole, Atul Kahate, “Operating Systems”, McGraw Hill Education, 2016.

Course Code	24EE01PR0402				
Category	Programme Core Course				
Course Title	Operating System Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

List of Experiments

Experiment 1: Basic OS Commands

Objective: Familiarize students with basic OS commands and their usage.

Tasks:

1. Navigate the filesystem using cd, ls, pwd, mkdir, rmdir.
2. Manage files using touch, rm, cp, mv.
3. View file contents with cat, more, less, head, tail.
4. Use file permissions and ownership commands: chmod, chown.

Experiment 2: Shell Scripting Basics

Objective: Introduce students to writing and executing basic shell scripts.

Tasks:

1. Write a script to display "Hello, World!".
2. Write a script to accept user input and display it.
3. Write a script to perform basic arithmetic operations.
4. Write a script to check if a file exists.
5. Write a script to display the current date and time.

Experiment 3: Process Management

Objective: Understand process creation, management, and termination.

Tasks:

1. Use commands ps, top, htop to monitor processes.
2. Create background and foreground processes using & and fg, bg.
3. Use kill, pkill, and killall to terminate processes.
4. Write a script to create child processes using fork() (C program).
5. Write a script to handle zombie and orphan processes (C program).

Experiment 4: Inter-process Communication (IPC)

Objective: Explore various IPC mechanisms.

Tasks:

1. Implement communication using pipes.
2. Use named pipes (FIFOs) for communication between unrelated processes.
3. Implement message passing using message queues.
4. Use shared memory for communication.
5. Implement synchronization using semaphores.

Experiment 5: Memory Management

Objective: Understand memory allocation and management.

Tasks:

1. Write a program to simulate first-fit, best-fit, and worst-fit memory allocation.
2. Implement paging and demonstrate page replacement algorithms (FIFO, LRU).
3. Use malloc(), calloc(), realloc(), and free() in C for dynamic memory allocation.
4. Monitor memory usage using commands like free, vmstat, top.

Experiment 6: File Systems and I/O Management

Objective: Explore file systems and basic I/O operations.

Tasks:

1. Use commands to mount and unmount file systems: mount, umount.
2. Explore file system hierarchy and structure.
3. Implement basic file operations (open, read, write, close) using system calls in C.
4. Write a program to simulate file allocation methods (contiguous, linked, indexed).
5. Monitor disk usage using df, du.

Experiment 7: Deadlock Detection and Avoidance

Objective: Understand deadlock concepts and implementation.

Tasks:

1. Write a program to simulate deadlock detection algorithm.
2. Implement the Banker's algorithm for deadlock avoidance.

Experiment 8: Scheduling Algorithms

Objective: Explore CPU scheduling algorithms.

Tasks: Implement various scheduling algorithm.

Course Code	24EE01TH0403			
Category	Programme Core Course			
Course Title	Design and Analysis of Algorithms			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:
On successful completion of the course, students will be able to:
1. Understand mathematical formulation, complexity analysis and methodologies to solve the recurrence relations for algorithms.
2. Apply Divide and Conquer algorithms and use them in examples
3. Formulate Greedy Methodology and use them in real life examples.
4. Design Dynamic programming and Backtracking Paradigms to solve real-life problems.
5. Design solutions using standard approaches comprehending NP class problems.

Syllabus:

Module I: (06 hrs)
Mathematical foundations for arithmetic and geometric series, Principles of designing algorithms and complexity calculation, Asymptotic notations for analysis of algorithms, worst case and average case analysis, amortized analysis and it's applications.

Module II: (06 hrs)
Divide and Conquer- Introduction to Divide and Conquer, Min Max Problem, Maximum sub-array problem, Closest pair of points problems, convex hull problem.

Module III: (06 hrs)
Greedy method – basic strategy, fractional knapsack problem, Minimum cost spanning trees, activity selection problem, find maximum sum possible equal to sum of three stacks.

Module IV: (06 hrs)
Dynamic Programming -basic strategy, Bellmen ford algorithm, all pairs shortest path, multistage graphs, optimal binary search trees, traveling salesman problem, Longest Common Subsequence problem.

Module V: (06 hrs)
Basic Traversal and Search Techniques, connected components, backtracking basic strategy, 8-Queen's problem, sum of subset problem, Introduction to Approximation algorithm.

Module VI: (05 hrs)
NP-hard and NP-complete - basic concepts, non-deterministic algorithms, NPhard and NP complete decision and optimization problems, polynomial reduction,vertex cover problem, clique cover problem.

Text Books

1. Thomas H. Cormen et.al; “Introduction to Algorithms”; 3 Edition; Prentice Hall, 2009.
2. Horowitz, Sahani and Rajasekaram; “Computer Algorithms”, Silicon Press, 2008.
3. Brassard and Bratley; “Fundamentals of Algorithms”, 1 Edition; Prentice Hall, 1995.

Reference Books

1. Parag Himanshu Dave, Balchandra Dave, “Design and Analysis of Algorithms” Pearson Education, O'relly publication
2. Richard Johnsonbaugh, “Algorithms”, Pearson Publication, 2003.

Course Code	24EE01TH0404			
Category	Programme Core Course			
Course Title	Fundamentals of AI and Machine Learning			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:

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

1. Analyze different AI problem-solving strategies, including search algorithms and game-playing techniques, to determine their effectiveness in various scenarios.
2. Perform exploratory data analysis to prepare datasets for machine learning models.
3. Implement and analyze supervised and unsupervised learning algorithms on a given data set.
4. Evaluate and interpret the performance of machine learning models using evaluation metrics and inference techniques to derive meaningful insights.
5. Apply appropriate AI and machine learning techniques to solve real-world problems by selecting suitable models, algorithms, and optimization strategies.

Syllabus:

Module 1: (5 Hrs)

Foundations of AI: Overview of AI and its applications, Intelligent Agents, Problem Solving: Uninformed and Informed Search, Local search, Constraint Satisfaction Problems, Game Trees: Minimax and Alpha-Beta Pruning.

Module 2: (6 Hrs)

Knowledge Representation and Reasoning:

Propositional and first-order logic, Inference methods and rule-based systems, Probabilistic Reasoning: Probability theory, Bayesian networks, and decision theory, Naive Bayes classifier; Bayes optimal classifiers, Maximum Likelihood Estimation, MAP;

Module 3: (7 Hrs)

Foundations for ML and EDA: Introduction to machine learning and its types, parametric vs non-parametric models, machine Learning pipeline. Exploratory Data Analysis- Data Cleaning, Handling Missing Values, Outlier Detection, Feature Selection Techniques, Feature Scaling and Transformation, Handling Imbalanced Data, Dimensionality Reduction (PCA)

Module 4: (9Hrs)

Supervised learning algorithms: Gradient Descent, Linear and Logistic Regression, Multivariate and Polynomial Regression – Bias/Variance Trade-off, overfitting and under fitting, Regularization. Decision Trees and Random Forests. Support Vector Machines, K-Nearest Neighbors.

Module 5: (8 Hrs)

Unsupervised learning & Model Evaluation: K-means clustering, Hierarchical Clustering, DBSCAN, Anomaly Detection: Isolation Forests, One-Class SVM.

Evaluation and Model Selection: Performance Metrics: Accuracy, Precision, Recall, F1-score, Confusion Matrix, ROC & AUC Curves, Evaluation Measures, Cross-Validation techniques.

Text Book:

1. Machine learning, by Mitchell Tom, First edition, McGraw Hill, 1997.

2. Artificial Intelligence: A Modern Approach by Stuart Russell, Peter Norvig, Third edition, Pearson, 2009.

Reference Books:

1. Introduction to Artificial Intelligence by Nikhil Sharma, Josh Hug, Jacky Liang, and Henry Zhu, University of California, Berkeley.

2. The Elements of Statistical Learning Data Mining, Inference, and Prediction by Trevor Hastie, Robert Tibshirani, Jerome Friedman, Second Edition, Springer, 2009.

3. Machine Learning: A Probabilistic Perspective by Kevin P. Murphy, Francis Bach; MIT Press, 2012.

4.

Course Code	24EE01PR0404				
Category	Programme Core Course				
Course Title	Fundamentals of AI and Machine Learning Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

List of Experiments:

Lab-01: **Implementing Search Algorithms for Problem Solving**

Lab-02: Knowledge Representation and Decision-Making using Bayesian Networks

Lab-03: Implement **data preprocessing techniques on the given dataset.**

Lab-04: Perform Dimensionality Reduction using Principal Components Analysis (PCA).

Lab-05: Implement linear regression algorithm (Single, Multiple variable and polynomial) using benchmark datasets and evaluate the performance of linear regression using evaluation measures.

Lab-06: Implement **the following algorithms to perform the task of classification on the** given datasets and evaluate the performance of algorithms using evaluation measures.

a) Logistic Regression

b) Decision Tree

c) Random Forest

d) K-nearest Neighbor

Lab-07: Build and implement **an image classifier using Support Vector machine (SVM) algorithm** and evaluate the performance of the trained model algorithms using k-fold cross-validation.

Lab-08: Build and develop a model for document classification using probabilistic machine learning algorithms.

Lab-09: Implement the K-means clustering algorithm to perform image segmentation and compare its performance with different numbers of clusters (k) using various evaluation metrics such as silhouette score, Davies-Bouldin index, and within-cluster sum of squares (WCSS).

Lab-10: Perform the comparative analysis of ensemble learning techniques on classification tasks.

Lab-11: **A Capstone Project:** Students are required to utilize the knowledge and competencies gained throughout the course to address a practical real-world challenge or investigate a substantial research query within the realm of machine learning.

Course Code	24EE01PR0407			
Category	Vocational and Skill Enhancement Course			
Course Title	Software Laboratory Practice-I			
Scheme & Credits	L	P	Credits	Semester
	0	2	1	IV

Course Objective:
The primary objective of this course is to provide hands-on experience in data visualization using Python. The course aims to equip students with the necessary skills to effectively represent data using various visualization techniques and libraries.

Course Outcomes (COs):
By the end of this course, students will be able to:
CO1: Understand the fundamental concepts of data visualization and its importance in data analysis.
CO2: Implement basic and advanced visualization techniques using Python libraries like Matplotlib, Seaborn, and Plotly.
CO3: Analyze and interpret real-world datasets using effective graphical representation.
CO4: Develop interactive and dynamic visualizations for better data-driven insights.
CO5: Build a complete data visualization project integrating multiple visualization techniques.

Syllabus: List of Experiments:

1. **Introduction to Data Visualization and Setup of Python Environment**
    a. Installing Python, Jupyter Notebook, and necessary libraries (Matplotlib, Seaborn, Plotly, Pandas).
    b. Introduction to basic plotting techniques.
2. **Creating Basic Charts Using Matplotlib**
    a. Implementing line plots, bar charts, scatter plots, and histograms.
    b. Customizing plots with labels, legends, colors, and markers.
3. **Exploratory Data Analysis (EDA) Using Pandas and Matplotlib**
    a. Loading datasets using Pandas.
    b. Visualizing distributions, trends, and relationships in data.
4. **Advanced Data Visualization Using Seaborn**
    a. Creating heatmaps, pair plots, violin plots, and swarm plots.
    b. Understanding categorical vs. numerical data visualization.
5. **Time Series Data Visualization Using Matplotlib and Seaborn**
    a. Plotting time-series data using line charts.
    b. Applying rolling statistics and trend analysis.

6. **Interactive Data Visualization Using Plotly**
    a. Creating interactive bar charts, line graphs, and scatter plots.
    b. Implementing hover effects and tooltips for better interactivity.
7. **Geospatial Data Visualization Using Folium and Plotly**
    a. Plotting geographical data using maps.
    b. Adding layers and markers to visualize spatial distributions.
8. **Multivariate Data Visualization for Correlation Analysis**
    a. Creating pair plots, correlation heatmaps, and joint plots.
    b. Understanding how multiple variables interact in datasets.
9. **Real-World Dataset Visualization and Storytelling with Data**
    a. Selecting a dataset and visualizing insights using multiple techniques.
    b. Creating a dashboard-like presentation of findings.
10. **Mini Project: Building a Data Dashboard Using Streamlit**
    a. Developing an interactive web-based dashboard for data visualization.
    b. Integrating different types of charts and user inputs.

Textbooks:

1. "Python Data Visualization Cookbook" – Igor Milovanovic, Dimitry Foures, Giuseppe Vettigli (Packt Publishing)
2. "Python for Data Analysis" – Wes McKinney (O'Reilly Media)
3. "Data Visualization with Python and JavaScript" – Kyran Dale (O'Reilly Media)
4. "Matplotlib for Python Developers" – Sandro Tosi (Packt Publishing)

Reference Books:

1. "Python Data Science Handbook" – Jake VanderPlas (O'Reilly Media)
2. "Data Visualization with Python: Create Stunning Charts and Plots" – Mario Döbler (Apress)
3. "Interactive Data Visualization with Python" – Abha Belorkar (BPB Publications)
4. "Hands-On Data Visualization with Bokeh" – Kevin Jolly (Packt Publishing)
5. "Seaborn Quick Start Guide" – Rahul Agarwal (Packt Publishing)

The **All-in-one software tool** for **Software Laboratory Practice-I (Data Visualization using Python)** course is: **Jupyter Notebook (Anaconda Distribution)**

Course Code	24EE01TH0405-01			
Category	Multidisciplinary Minor -2			
Course Title	MIPS Processor Design and Testing			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes:

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

1. Understand the architecture and working of a single-cycle and multi-cycle MIPS processor, including the data path and control path design.
2. Design individual components of the MIPS processor in Verilog HDL and integrate them into a complete processor.
3. Implement and simulate the MIPS data path and control unit in Verilog HDL on FPGAs.
4. Develop testbenches in Verilog HDL to verify the functionality of individual modules and the integrated processor.
5. Perform step-by-step debugging and testing of a MIPS processor using waveform analysis and behavioral simulation.

Syllabus:

Module-I: Introduction to MIPS Processor and Design Flow

Introduction to computer system and its sub modules, Introduction to RISC and CISC paradigm, overview of the MIPS Instruction Set Architecture (ISA), registers, instruction formats, and addressing modes, single-cycle and multi-cycle execution, breakdown of the MIPS data path into instruction fetch, decode, execute, memory, and writeback stages.

Module-II: Designing the MIPS ALU and Register File

Hardware modeling for Arithmetic and logical operations performed by the MIPS ALU, implementation of a register file, testbench creation for verifying the ALU and register file operations.

Module-III: Instruction Fetch and Decode Stage Implementation

The instruction fetch stage, design of program counter (PC) and instruction memory components, instruction decode stage, generation of control signals, Integration of the
instruction memory with the register file.

Module-IV: Execution, Memory, and Writeback Stages

Implementation of the execution stage, ALU operation and branching logic, memory access stage, design of control logic for load store instructions, implementation of write back stage, testing of R-type, I-type, and J-type instructions using data and control path.

Module-V: Integration and Testing of Single-Cycle MIPS Processor

Assembling the complete single-cycle MIPS processor, develop a complete Verilog testbench for the

entire processor, run test programs, and debug issues that arise during simulation, analyze timing diagrams and waveforms using simulation tools.

Module-VI

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

Text Books:

1. Computer Organization and Design Edition - The Hardware/Software Interface, David A. Patterson, John L. Hennessy, 5th Edition, 2014.

Reference Books

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

Course Code	24EE01TH0405-02			
Category	Multidisciplinary Minor -2			
Course Title	Programming for Environmental IoT			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Outcomes

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

Module 1: Sensors for Environmental Monitoring

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

Module 2: Actuators for Environmental Monitoring

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

Module 3: Interfacing with Raspberry Pi

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

Module 4: Interfacing with ESP8266

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

Module 5: Displaying Sensor Data Remotely

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

Text Books:

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

Reference Books:

1. Internet of Things Principles, Paradigms and Application of IoT, Joseph Kofi Wireko, Kaml Hiran, BPB Publications 2020 1st Edition

2. Raspberry Pi User Guide, EbenUptonand Gareth Halfacree, John Wiley& Sons, 2016.

Course Code	24SM01TH0401				
Category	HSSM				
Course Title	Innovation and Entrepreneurship				
Scheme & Credits	L	T	P	Credits	Semester
	1	0	0	1	IV

Course Outcomes

CO 1: Understand the fundamental concepts of innovation and entrepreneurship.
CO 2: Learn about business idea generation, startup processes, and financial planning.

UNIT :1 Fundamentals of Innovation and Entrepreneurship: Definition, importance, and scope of entrepreneurship**,** mindset and characteristics of entrepreneurs**,** Innovation: Types, sources, and the innovation process**,** Role of technology and digital transformation in entrepreneurship
UNIT :2 Business Idea, Financial Planning, and Growth Strategies: Identifying and evaluating business opportunities**,** Business model Canvas, Market research and customer validation**,** Funding sources (venture capital, angel investment, government schemes)**,** Business sustainability and growth strategies

Text Books:

Textbook:

- **"Innovation and Entrepreneurship" – Peter F. Drucker** (HarperBusiness)

Reference Books:

- **Robert D. Hisrich** - *Entrepreneurship*, Tata McGraw-Hill
- **Vasant Desai** - *Dynamics of Entrepreneurial Development and Management*, Himalaya Publishing House
- **S.S. Khanka** - *Entrepreneurial Development*, S. Chand & Co.
- **Paul Trott** - *Innovation Management and New Product Development*, Pearson

Assignments:

1. **Case Study on an Entrepreneur**: Choose an entrepreneur and analyze their journey, challenges, and success factors.
2. **Business Plan Development**: Develop a business plan, including market research, financial projections, and risk analysis.

Course Code	24SM01PR0401				
Category	HSSM				
Course Title	Innovation and Entrepreneurship				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

Course Outcomes:

CO 1: Apply innovation and entrepreneurship principles through hands-on projects.
CO 2: Develop problem-solving and business idea implementation skills.

UNIT :1 Ideation and Prototyping: Brainstorming and idea generation exercises**,** Design thinking methodology for problem-solving**,** Creating a prototype or minimum viable product (MVP)**,** Validating business ideas through customer feedback

UNIT :2 Startup Execution and Market Strategies: Business model canvas workshop**,** Digital marketing and branding basics for startups**,** Pitching an idea to investors or stakeholders**,** Developing an innovation-driven business strategy

Text Books:

Textbook:

- **"Innovation and Entrepreneurship" – Peter F. Drucker** (HarperBusiness)

Reference Books:

- Eric Ries - *The Lean Startup*, Crown Publishing
- Ash Maurya - *Running Lean: Iterate from Plan A to a Plan That Works*, O'Reilly Media
- Guy Kawasaki - *The Art of the Start 2.0*, Portfolio
- Alexander Osterwalder & Yves Pigneur - *Business Model Generation*, Wiley

Assignments:

3. **Problem-Solving Hackathon**
    - Work on a real-world problem and propose an innovative business solution.
4. **Business Model Canvas Development**
    - Create and present a business model canvas for a startup idea.
5. **Pitch Deck Presentation**
    - Develop and present a startup pitch deck in front of a panel.
6. **Prototype Showcase**

- Design and demonstrate a basic working prototype or proof of concept.

Course Code	24EE01HT0301				
Category	Honors (Full-Stack JavaScript Developer)				
Course Title	JavaScript Programming Essentials				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	III

Course Outcomes:

1. Understand the role of JavaScript in modern web development and its integration with HTML and CSS.
2. Apply JavaScript concepts to create dynamic and interactive front-end user interfaces.
3. Analyze different JavaScript frameworks, libraries, and tools to enhance web applications.
4. Propose solutions for real-world problems using JavaScript programming and front-end-backend integration.
5. Evaluate the scalability and performance of JavaScript-based web solutions through optimized coding practices.

Syllabus

Module 1: Introduction to JavaScript Programming Environment, Core Syntax, Variables, Data Types, Control Structures, and Functions.

Module 2: Document Object Model (DOM) Manipulation and Event Handling, Introduction to APIs and Fetch Requests.

Module 3: Front-end Components: JavaScript Libraries (e.g., jQuery) and Frameworks (e.g., React Basics), UI Design, and Animation Techniques.

Module 4: Backend Integration: JavaScript and Node.js Overview, Data Handling with JSON, Working with REST APIs.

Module 5: Use Cases of JavaScript Algorithms: Problem Solving in Web Applications, Full-stack Implementation Scenarios using Express.js and MongoDB.

Text Book:

1. Eloquent JavaScript: A Modern Introduction to Programming, Marijn Haverbeke, No Starch Press, 3rd Edition (2018).

Reference Books:

1. JavaScript: The Definitive Guide, David Flanagan, O'Reilly Media, 7th Edition (2020).
2. JavaScript and JQuery: Interactive Front-End Web Development, Jon Duckett, Wiley (2014).
3. MDN Web Docs: JavaScript Reference and Tutorials.

Course Code	24EE01HT0401			
Category	Honors (Full-Stack JavaScript Developer)			
Course Title	Front and Back-End App Development			
Scheme & Credits	L	P	Credits	Semester
	3	0	3	IV

Course Objective:
The primary goal of this course is to equip students with the skills required to develop full-stack web applications by integrating both front-end and back-end technologies. The course focuses on building interactive, responsive, and scalable applications using modern web development frameworks and tools.

Course Outcomes (COs) for Theory and Lab:
By the end of this course, students will be able to:
CO1: Understand and apply full-stack development concepts using modern frameworks and technologies.
CO2: Develop interactive and dynamic front-end applications using React.js with state management and routing.
CO3: Design and implement back-end solutions using Node.js and Express.js, including RESTful API development.
CO4: Integrate and manage a NoSQL database using MongoDB and Mongoose ORM for efficient data handling.
CO5: Deploy, test, and optimize full-stack web applications using cloud-based platforms and industry best practices.

Syllabus:

Module 1: Introduction to Full-Stack Development (9 Hours)
Overview of Front-End, Back-End, and Full-Stack Development, Understanding the MERN (MongoDB, Express.js, React.js, Node.js) and MEVN (MongoDB, Express.js, Vue.js, Node.js) stacks, Setting up a development environment (Node.js, npm, Git, VS Code), Introduction to RESTful APIs and MVC architecture.

Module 2: Front-End Development with React.js (10 Hours)
Understanding React.js fundamentals (Components, Props, State, JSX), React Hooks and Functional Components, React Router for navigation and single-page applications (SPA), Styling in React (CSS Modules, Styled Components, Tailwind CSS), API integration in React using Fetch and Axios.

Module 3: Back-End Development with Node.js and Express.js (9 Hours)
Introduction to Node.js (Event Loop, Async/Await, Modules), Building RESTful APIs using Express.js, Middleware and Routing in Express.js, Connecting to a database (MongoDB with Mongoose ORM), User authentication with JWT (JSON Web Tokens).

Module 4: Database Management with MongoDB (9 Hours)
Understanding NoSQL vs SQL databases, CRUD operations in MongoDB, Designing database schemas using Mongoose, Aggregation and indexing in MongoDB, Cloud-based database solutions

(MongoDB Atlas).

Module 5: Deployment, Testing, and Optimization (8 Hours)
Introduction to DevOps for full-stack applications, Deploying front-end apps (Vercel, Netlify), Deploying back-end apps (Heroku, Render), Testing APIs with Postman, Debugging and performance optimization in full-stack applications.

Lab Syllabus:

1. Setting Up Full-Stack Development Environment – Installing Node.js, npm, Git, VS Code, and MongoDB. Initializing a full-stack project using Express.js.
2. Building a Simple React.js Application – Creating a React app with functional components, props, and state management. Implementing event handling and form submission.
3. Implementing Routing in React.js Using React Router – Creating multiple pages with navigation using React Router. Implementing dynamic routing and route parameters.
4. Creating RESTful APIs with Express.js and Node.js – Developing a REST API with GET, POST, PUT, DELETE routes. Implementing middleware for request handling.
5. Connecting React Front-End to Node.js Back-End – Fetching and displaying data in React from a backend Express API using Axios. Handling API responses and errors.
6. Database Integration with MongoDB and Mongoose – Setting up MongoDB with Mongoose. Performing CRUD operations and designing database schemas.
7. User Authentication and JWT-Based Authorization – Implementing user login, registration, and authentication using JWT. Protecting routes with authentication middleware.
8. Deploying a Full-Stack Web Application – Hosting the front-end on Vercel/Netlify and the back-end on Heroku/Render. Connecting both for a live application.

Textbooks:

1. **"Full-Stack React, TypeScript, and Node"** – David Choi, Packt Publishing
   - Covers full-stack development using React, Node.js, Express, and MongoDB.
2. **"Learning React: Modern Patterns for Developing React Apps"** – Alex Banks, Eve Porcello, O'Reilly Media
   - A comprehensive guide to React.js with hooks, state management, and modern patterns.
3. **"Node.js Design Patterns"** – Mario Casciaro, Luciano Mammino, Packt Publishing
   - Explains best practices and design patterns for building scalable back-end applications.
4. **"MongoDB: The Definitive Guide"** – Shannon Bradshaw, Eoin Brazil, Kristina Chodorow, O'Reilly Media
   - Covers MongoDB fundamentals, database design, and advanced NoSQL concepts.

5. **"Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node"** – Vasan Subramanian, Apress
    - A hands-on guide for building full-stack applications using the MERN stack.

Reference Books:

1. **"JavaScript: The Definitive Guide"** – David Flanagan, O'Reilly Media
    - In-depth coverage of JavaScript fundamentals, ES6+, and modern JavaScript features.

2. **"Eloquent JavaScript"** – Marijn Haverbeke, No Starch Press
    - A beginner-friendly book covering JavaScript, functional programming, and asynchronous programming.

3. **"Web Development with MongoDB and Node"** – Ethan Brown, O'Reilly Media
    - A practical guide for integrating MongoDB with Node.js and Express.js.

4. **"React Up & Running"** – Stoyan Stefanov, O'Reilly Media
    - Provides a fast-paced introduction to React and its ecosystem.

5. **"The Road to React"** – Robin Wieruch
    - A hands-on approach to learning React with real-world projects.

Course Code	24EE01MT0301				
Category	Minor Course				
Course Title	IoT fundamentals				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	III

Course Outcomes

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

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

Syllabus

Module - I: (7Hrs)

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

Module - II: (7Hrs)

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

Module - III: (7Hrs)

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

Module - IV: (7Hrs)

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

Module - V: (7Hrs)

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

Text Books

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

Course Code	24EE01MT0401				
Category	Minor Course				
Course Title	Sensor Interfacing with Arduino and ESP8266				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

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

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

Syllabus

Module I: (6Hrs)
Introduction to Arduino Prototyping Platform: Arduino IDE, Arduino C, Setting up the Arduino board, creating sketches,using Libraries, using example codes, Debugging Using the Serial Monitor.

Module II: (7Hrs)
Sensor interfacing with Arduino: Analog and digital sensors, Temperature sensors, Humidity sensors, Obstacle sensors, Ultrasonic sensor, Accelerometer and gyro, etc.

Module III: (7Hrs)
Serial Communication in Arduino: Serial and parallel communication, Serial communication protocols, UART, I2C, SPI, Wired and Wireless communication, Interfacing Communication Modules with Arduino.

Module IV: (7 Hrs)
Interfacing Displays and Actuators: 16x2 LCD, Graphical LCD, Graphical OLED, Arduino Tone functions, Melody generation on a piezo buzzer, Speed and direction control of DC, Servo and, Stepper Motor.

Module V: (6Hrs)
Introduction to ESP8266: ESP8266 development board, Programming ESP8266 through Arduino IDE, connecting to the internet, sending and receiving data on internet.

Module VI: (7Hrs)
Interfacing sensors and actuators with ESP8266: LDR, Temperature sensor, Humidity sensor, IR sensor, OLED, RGB LED, Servo motor, etc.

Text Books:

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