Pandit Deendayal Energy University

School Of Technology

Department of Computer Science and Engineering

Under Graduate Curriculum Handbook (Academic Year 2024-28)

B.Tech. (Computer Engineering)
w. e. f. July, 2024.

Vision

"To contribute to the society by imparting transformative education and producing globally competent professionals having multidisciplinary skills and core values to do futuristic research & innovations."

Mission

- To accord high quality education in the continually evolving domain of Computer Engineering by offering state-of-the-art undergraduate, postgraduate, doctoral programmes.
- To address the problems of societal importance by contributing through the talent we nurture and research we do:
- To collaborate with industry and academia around the world to strengthen the education and multidisciplinary research ecosystem.
- To develop human talent to its fullest extent so that intellectually competent and imaginatively exceptional leaders can emerge in a range of computer professions.

Program Educational Objectives (PEOs)

PEO-1. To prepare graduates who will be successful professionals in industry, government, academia, research, entrepreneurial pursuit and consulting firms

PEO-2. To prepare graduates who will make technical contribution to the design, development and production of computing systems

PEO-3. To prepare graduates who will get engage in lifelong learning with leadership qualities, professional ethics and soft skills to fulfill their goals

PEO-4. To prepare graduates who will adapt state of the art development in the field of computer engineering

Program Outcomes (POs)

PO-1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO-2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO-3. Design / development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO-4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO-5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO-6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO-7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO-8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO-9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO-10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO-11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO-12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO-1. Develop computer engineering solutions for specific needs in different domains applying the knowledge in the areas of programming, algorithms, hardware-interface, system software, computer graphics, web design, networking and advanced computing.

PSO-2. Analyze and test computer software designed for diverse needs.

PSO-3. Pursue higher education, entrepreneurial ventures and research.

Course Outline

Sem		Course Name	Th	Tut	Pra	Hrs	Cr
Sem I	HSC	Humanities – I	2	0	0	2	2
	BSC	Mathematics – I	3	1	0	4	4
	BSC	Applied Physics	3	0	0	3	3
	BSC	Applied Physics Laboratory	0	0	2	2	1
	BSC	Environment Science	2	0	0	2	2
	ESC	Workshop Practices	0	0	2	2	1
	BSC	Biology for Engineers	2	0	0	2	2
	ESC	Computer Programming -I	1	0	0	1	1
	ESC	Computer Programming -I Laboratory	0	0	2	2	1
	HSC	Universal Human Values	1	0	0	1	1
	HSC	Indian Knowledge System	2	0	0	2	2
			16	**1**	**6**	**23**	**20**
Sem II	HSC	Humanities – II	2	0	0	2	2
	BSC	Mathematics – II	3	1	0	4	4
	BSC	Engineering Chemistry	3	0	0	3	3
	BSC	Engineering Chemistry Laboratory	0	0	2	2	1
	ESC	Elements of Electrical & Electronics Engineering	3	0	0	3	3
	ESC	Elements of Electrical & Electronics Engineering Laboratory	0	0	2	2	1
	ESC	Engineering Graphics	0	0	4	4	2
	HSC	Yoga, Health & Hygiene OR NCC/NSS	0	0	2	2	1
	HSC	Organizational Behavior	1	0	0	1	1
	ESC	Computer Programming -II	1	0	0	1	1
	ESC	Computer Programming -II Laboratory	0	0	2	2	1
			13	**1**	**12**	**26**	**20**
Sem III	Pro	Civic and Social Service Internship	0	0	0	0	1
	BSC	Discrete Mathematics	3	1	0	4	4
	PC	Database Management System	3	0	0	3	3
	PC	Database Management System Laboratory	0	0	2	2	1
	PC	Digital Logic and Design	3	1	0	4	4
	PC	Data Structures	3	0	0	3	3
	PC	Data Structures Laboratory	0	0	2	2	1
	PC	Object Oriented Programming	3	0	0	3	3
	PC	Object Oriented Programming Laboratory	0	0	2	2	1
			15	**2**	**6**	**23**	**21**
Sem IV	OE	Open Elective 1 (From Other School) Open-Source Technologies Introduction to Data Science	3	0	0	3	3
	PC	Design Thinking	0	0	2	2	1
	PC	Probability and Statistics Theory	3	0	0	3	3
	PC	Computer Organization and Architecture	3	1	0	4	4
	PC	Software Engineering	3	0	0	3	3

	PC	Software Engineering Laboratory	0	0	2	2	1
	PC	Theory of Computation	3	1	0	4	4
	PC	Design and Analysis of Algorithm	3	0	0	3	3
	PC	Design and Analysis of Algorithm Laboratory	0	0	2	2	1
			18	**2**	**6**	**24**	**23**
Sem V	OE	Open Elective 2 ((NPTEL/SWAYAM/MOOC))	3	0	0	3	3
	ESC	Introduction to Artificial Intelligence (Department Specific)	3	0	0	3	3
	HSC	Engineering Economics	3	0	0	3	3
	PE	Program Elective 1	3	0	0	3	3
	PC	Computer Networks	3	0	0	3	3
	PC	Computer Networks Laboratory	0	0	2	2	1
	PC	Compiler Design	3	0	0	3	3
	PC	Compiler Design Laboratory	0	0	2	2	1
	PC	Operating System	3	0	0	3	3
	PC	Operating System Laboratory	0	0	2	2	1
			21	**0**	**6**	**27**	**24**
Sem VI	OE	Open Elective 3 (From Other Department of FoET) Web Development Object Oriented Programming & Design	3	0	0	3	3
	PE	Program Elective 2	3	0	0	3	3
	PE	Program Elective 3	3	0	0	3	3
	PC	Cryptography and Network Security	3	0	0	3	3
	PC	Cryptography and Network Security Laboratory	0	0	2	2	1
	PC	Web & Mobile Development Essentials	1	0	0	1	1
	PC	Web & Mobile Development Essentials Laboratory	0	0	4	4	2
	PC	Distributed Computing	3	0	0	3	3
	PC	Distributed Computing Laboratory	0	0	2	2	1
			16	**0**	**8**	**24**	**20**
Sem VII	Pro	Summer Internship	0	0	0	0	2
	OE	Open Elective 4 (From Other Department of FoET) Green IT Global Data Security & Privacy Laws	3	0	0	3	3
	ESC	Industry 4.0 (Department Specific)	2	0	0	2	2
	ESC	Industry 4.0 Laboratory (Department Specific)	0	0	2	2	1
	PE	Program Elective 4	3	0	0	3	3
	PE	Program Elective 5	3	0	0	3	3
	PC	Machine Learning	3	0	0	3	3
	PC	Machine Learning Laboratory	0	0	2	2	1

	PC	Cyber Laws and Ethics	1	0	0	1	1
	Pro	Seminar					1
			15	**0**	**4**	**19**	**20**
Sem		**Course Name**	**Th**	**Tut**	**Pra**	**Hrs**	**Cr**
VIII	**Pro**	Major/Comprehensive Project					**12**

Total Credits: 160

List of Electives (Track-based)

	Artificial Intelligence/ Machine Learning Track	**Industry Track**	**Speech, Vision & Text Track**	**Computing Track**	**Communication Track**
PE-1	Data Mining and Data Warehousing	Object Oriented Modelling and Design	Computer Graphics	Advanced Data Structure and Algorithms	Data Communication
PE-2	Soft Computing	UI/UX Design	Digital Image Processing	Blockchain Technology	Mobile Computing
PE-3	Deep Learning	Secure Software Engineering	Computer Vision	Big Data Analytics	Wireless Sensor Networks
PE-4	Agent based Learning	Agile and DevOps	Natural Language Processing	Cloud Computing	Internet of Things
PE-5	Machine Learning in Cyber Security	Web Application Testing	Speech Processing	Quantum Computing	Autonomous Systems

1st Semester

<Course Code>				Mathematics – I						
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	--	--	100

COURSE OBJECTIVES

- To be able to evaluate problems related to differential and integral calculus of complex functions.
- To be able to obtain area, volume using integral calculus.
- To be able to formulate and solve various engineering problems using the calculus.
- To study the properties of sequence and series and to check the convergence and divergence.

UNIT I: DIFFERENTIAL CALCULUS AND ITS APPLICATIONS **08 Hrs.**

Partial derivative and its application - Euler's theorem - Total derivatives - Jacobians – Maxima and Minima of two variables using Lagrange's multipliers. Taylor's series, Convergence of power series.

UNIT II: INTEGRAL CALCULUS AND ITS APPLICATIONS **12 Hrs.**

Definition and evaluation of double integral (Cartesian – Polar form) – Change of orders - Change of variables – Evaluation of triple integral, change of variables (Cartesian to spherical – and cylindrical) – Applications, area – volume – center of mass – center of gravity by double and triple integral.

UNIT III: VECTOR CALCULUS **10 Hrs.**

Gradient, divergence and curl – Directional derivative – Irrotational and Solenoidal vector fields – Vector Integration – Simple problems on line, surface and volume integrals – Green's theorem in a plane, Gauss divergence theorem and Stokes' theorem (without proofs) – Simple application involving cubes and rectangular parallelepipeds.

UNIT IV: SEQUENCE AND SERIES **12 Hrs.**

Definition. Convergent, divergent, bounded & monotone sequences. Infinite sums: Basics Taylor series, Convergence and divergence, Tests for convergence for positive term series, Alternating series – Leibnitz test, Absolute convergence, conditional convergence.

TOTAL HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : Identify the use of convergence of infinite series in engineering aspects.
CO2 : Understand the concept of Directional derivative, Irrotational and Solenoidal vector fields.
CO3 : Apply the concept of differential and integral calculus in engineering problems.
CO4 : Analyze the obtained solution in linear and non-linear domains.
CO5 : Appraise mathematical problems from complex domain.
CO6 : Evaluate problems on Green's, Stokes' and Divergence theorems.

TEXT/REFERENCE BOOKS

1. B. S Grewal, Higher Engineering Mathematics, Khanna Pub.
2. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, Alpha Science.
3. Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley.
4. G. Strang, Linear Algebra and its Applications, Cengage Learning.
5. K. Hoffman and R. A. Kunze, Linear Algebra, Prentice Hall of India.

<Course Code>					Applied Physics (For CS, ICT, ECE, EE)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Understand Electric and Magnetic Fields, applying vector algebra.
- Apply Maxwell's equations to analyse electromagnetic waves and transmission.
- Analyse semiconductor behaviour and its applications in electronic devices
- Evaluate optical phenomena and their engineering applications in communications.

UNIT I: ELECTRICITY AND MAGNETISM **12 Hrs.**

Vector Algebra, Fundamental theorems of Gradient, Curls, and Divergence, Curvilinear co-ordinates, Continuous charge distribution, Divergence and Curl of Electrostatic Field, Electric potential and its applications, Work and Energy in Electrostatic, Bio-Savart's law, Divergence and curl of magnetic fields, Vector Potential, Ohm's law, EMF, Faraday's law of electromagnetic induction, Energy in Magnetic Fields, Maxwell's correction to ampere's law and Maxwell's equations, Poynting Vector.

UNIT II: ELECTROMAGNETIC WAVES **10 Hrs.**

Waves equation, Reflection and Transmission of waves, Polarisation, Wave equation for E and B for monochromatic plane waves, Propagation in linear media, reflection and transmission in normal and oblique incidence, Electromagnetic waves in conductors, Frequency dependence of permittivity, Waveguides, TE waves in rectangular waveguide, The Coaxial transmission line.

UNIT III: PHYSICS OF SOLIDS **10 Hrs.**

Fermi electron gas, Fermi level and surface, Energy bands, Energy Gap, Energy and band structure of conductor, insulators and semiconductors, Intrinsic semiconductors at 0K and room temperature, Intrinsic conductivity, Types of semiconductors, doping impurities, Temperature variation of carrier concentration, Electrical conductivity in semiconductors, Hall Effect, and magnetic materials

UNIT IV: OPTICS **10 Hrs.**

Nature of light waves, Fermat's principle, Coherent Sources, Interference, Two source interference, Interference in thin films, Newton's ring, Fresnel and Fraunhofer diffraction, Diffraction from single slit and double slit, Lasers, optical fibres and Holography, Applied optics: engineering measurements.

TOTAL HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : Recall fundamental laws and principles of electromagnetism and optics.
CO2 : Explain concepts of waves, conductivity, and semiconductor properties.
CO3 : Utilize principles to solve problems in electricity, magnetism, and optics.
CO4 : Evaluate electromagnetic phenomena and semiconductor behaviour through experimentation.
CO5 : Assess the effectiveness of engineering design for electromagnetic and optical devices.
CO6 : Design solutions for engineering challenges involving electromagnetics and semiconductors.

TEXT/REFERENCE BOOKS

1. Griffith, D. J., Introduction to Electrodynamics, Prentice Hall.
2. M. N. Avadhanulu and P G Kshirsagar, A text book of Engineering Physics, S.Chand Publications.
3. Sears and Zemansky, University physics, Pearson publications.
4. Principles of Electromagnetics, Matthew N. O. Sadiku, Oxford publications.
5. Hecht, E., Optics, Pearson Education.
6. M. A. Wahab, Solid State Physics: Structure and Properties of Materials, Narosa Publishing House Pvt. Ltd.-New Delhi.

<Course Code>					Modern Physics (For CH, PE, BT)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Understanding concepts of modern physics.
- Explaining the physics of EM waves and its propagation.
- Applying the Acquired basic knowledge of solid state physics.
- Analyze the Concepts of nuclear radiation physics.
- Estimate and correlate the concepts learned so far for various engineering applications.
- Apply the fundamentals designs of modern physics to solve complex physical problems.

UNIT I: MODERN PHYSICS **09 Hrs.**

Review of quantum concepts: particle nature of light, photoelectric effect, Compton effect, matter waves, wave packets, phase and group velocity, Interference, Diffraction and Polarization, Engineering Physics related applications. Davisson Germer experiment, Heisenberg uncertainty principle.

UNIT II: ELECTROMAGNETIC WAVES **12 Hrs.**

Physical and mathematical concepts of gradient, divergence and curl, Gauss theorem, applications in gravitation and electrostatics. Stokes' theorem and engineering Physics related applications. Equation of continuity, Biot Savart law – Ampere's law – magnetization and magnetic intensity, Faraday's law of Maxwell's equations, wave equation for electromagnetic radiation, applications of optical fibers in communication.

UNIT III: SOLID STATE PHYSICS **12 Hrs.**

Crystalline and Amorphous Solids, Ionic Crystals, Covalent Crystals, Van der Waals Bond, Band Theory of Solids, Semiconductor Devices, Electrical conductivity, Resistivity, Magnetism, Superconductivity, Introduction to BCS Theory.

Concepts of LASER, Interaction of radiation of matter-quantum mechanical view, characteristics and Types of laser, Engineering Physics related application of lasers.

UNIT IV: NUCLEAR RADIATION PHYSICS **09 Hrs.**

Mass defect, binding energy, Radioactivity, Types of Radiation, Interaction of Radiation with matter, Radiation detector, nuclear reactions, elements of nuclear reactors, fission and fusion, Engineering Physics related problems.

HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : To relate to shape the engineering perspective in the student mind.
CO2 : Aims to provide an understanding to analyse the physical phenomena of various physics concepts.
CO3 : To develop an analytical perspective in the student.
CO4 : To enable understanding in the students the importance of application of already studied topics.
CO5 : To explain and relate the importance of interdisciplinary problems.
CO6 : To strengthen problem solving attitude in physics using mathematical tools.

TEXT/REFERENCE BOOKS

1. Resnick, Halliday and Krane, Physics part I and II, John Wiely
2. Ghatak, Optics, Tata McGraw Hill
3. Purcell E.M. Electricity and Magnetism - Berkeley Physics Course, Vol.2, Tata McGraw-Hill.
4. Kittel C., Knight W.O. and Ruderman M.A., Mechanics - Berkeley Physics Course, Tata McGraw-Hill.
5. Griffith D.J.H., Introduction to Electrodynamics - Prentice Hall, India.
6. M. N. Avadhanulu, A text book of engineering Physics, S. Chand & Company, Ltd.
7. Feyman R.P., Leighton R.B. and Sands M. The Feyman Lectures on Physics, Vol. 1., Narosa Publication.

<24XXXXXX>					Engineering Physics					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To master the concepts of vector algebra and the fundamentals of mechanics.
- To explore the principles of elasticity in solids, band theory of solids and the Physics of semiconductors.
- Investigate the behavior of waves and oscillations, along with their applications in acoustics and ultrasonics.
- Gain insights into thermal physics, covering conduction, convection, and radiation, and their practical applications.

UNIT I: MECHANICS **12 Hrs.**

Introduction to vector algebra, Vector fields and scalar fields, Concepts of gradient, Divergence and Curl, Gauss-Divergence Theorem, Stokes' theorem and its applications, The everyday forces of Physics, Work-energy theorem, Rigid body motion and the conservation of angular momentum, Center of mass, Center of gravity, Moment of inertia: Definition, Moment of inertia of simple planar laminas.

UNIT II: PHYSICS OF SOLIDS **10 Hrs.**

Elasticity, Stress & Strain, Hook's law, Young's modulus, Poisson's ratio, Rigidity modulus, Relationship between elastic constants, Properties of solids, electrical conductivity, Wiede-mann Franz Law, Band structure of conductor, insulator and semiconductor, Intrinsic and extrinsic semiconductor, Conductivity of semiconductors, charge concentrations, Hall effect, LED and Solar cells, Magnetism and its origin, Magnetic properties of solids, Physics of superconductors, Applications.

UNIT III: WAVES, OSCILLATIONS AND ACOUSTICS **08 Hrs.**

Waves and oscillations: Types of waves, Simple harmonic motion, Damped simple harmonic motion, types of damping, Forced oscillation, Resonance, Energy Transport in Wave motion. Acoustics & Ultrasonic: Introduction to Sound, Sabine's reverberation theory, Acoustical defects and their remedies, Doppler Effect, Ultrasonic waves, Methods of their generation and detection, Properties and applications of ultrasonic waves.

UNIT IV: THERMAL PHYSICS **12 Hrs.**

Thermodynamic systems (closed and open), Thermodynamic properties and equilibrium, Concepts of heat and work, Laws of thermodynamics, Carnot cycle, Heat transfer-thermal expansion of solids and liquids, Conduction in solids, Thermal conductivity, Forbe's method, Lees' disc method, Thermal insulation and its applications, Thermal Convection and its applications, Newton's law of cooling, Thermal Radiation – emission and absorption radiation, emissive power, Black body radiation, Stefan's laws, Wien's law.

HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : Describe fundamental concepts of mechanics, such as vector algebra, everyday forces of Physics, Moment of inertia, etc
CO2 : Explain the principles underlying the physics of solids, including elasticity and the behavior of energy bands.
CO3 : Solve problems involving mechanics, solids, waves, oscillations, and acoustics
CO4 : Analyze the behavior of waves, oscillations, and thermal phenomena, including their practical applications
CO5 : Critically evaluate the significance of thermodynamic laws and superconductivity theories in solving real-world problems.
CO6 : Develop solutions and innovative applications in mechanics, solids, waves, and thermal physics by integrating concepts and principles learned throughout the course.

TEXT/REFERENCE BOOKS

1. D. Kleppner and R. J. Kolenkow, An Introduction to Mechanics, Tata McGraw-Hill.
2. D. Halliday, R. Resnick, J. Walker, Fundamentals of Physics, John Wiley & Sons.
3. Sears and Zemansky, University Physics, Pearson publications.
4. M. N. Avadhanulu, P.G. Kshirsagar, A textbook of Engineering Physics, S Chand & Co. Ltd.
5. F. M. A. Wahab, Solid State Physics: Structure and Properties of Materials, Narosa Publishing House Pvt. Ltd
6. Richard, Oscillations and Waves: An Introduction, Taylor & Francis.
7. Y. A. Cengel, Introduction to Thermodynamics and Heat Transfer, Tata McGraw-Hill.
8. B. Lal and N. Subramaniyam, Heat and Thermodynamics, S Chand & Co. Ltd.

<Course Code>					Applied Physics Laboratory (For CS, ICT, ECE, EE)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- Understand principles of electromagnetism and their experimental applications.
- Analyse experimental setups and procedures related to electricity and magnetism.
- Apply concepts of electromagnetic phenomena and optics in practical experiments.
- Investigate semiconductor device electrical properties through experimentation.

LIST OF EXPERIMENTS

1 To determine e/m using Thomson's method.
2 To study Bio-Savart's Law.
3 To verify Faraday and Lenz's law.
4 To study the magnetic field along the axis of a coil
5 To determine the electrical conductivity of metals.
6 To study the characteristics of Si solar cells.
7 To study the phenomenon of photoconductivity using CdS photo-resistor.
8 To determine energy band gap of semiconductor using four probe method.
9 To study the hall effect and determine hall voltage, hall coefficient, type of majority charge carriers, carrier concentration and hall angle.
10 To study of the ferromagnetic hysteresis.
11 To determine the wavelength of monochromatic light (sodium light) using Newton's rings.
12 To measure the slit width of single, blade and double slits.
13 To understand fundamental of optical fibres and analogue optical fibre communication.

COURSE OUTCOMES

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

CO1 : Recognize electromagnetic principles and practical applications in experimental setup.
CO2 : Interpret experimental setups and procedures for electricity, magnetism, and optics.
CO3 : Demonstrate and implement the concepts of electromagnetic phenomena and optics.
CO4 : Investigate the electrical properties of semiconductor device.
CO5 : Examine the experimental data to identify trends and physical relationships.
CO6 : Design experiments to investigate electromagnetism and semiconductor properties.

TEXT/REFERENCE BOOKS

1. Griffith, D. J., Introduction to Electrodynamics, Prentice Hall.
2. M. N. Avadhanulu and P G Kshirsagar, A text book of Engineering Physics, S. Chand Publications.
3. Sears and Zemansky, University physics, Pearson publications.
4. Principles of Electromagnetics, Matthew N. O. Sadiku, Oxford publications.
5. Hecht, E., Optics, Pearson Education.
6. M. A. Wahab, Solid State Physics: Structure and Properties of Materials, Narosa Publishing House Pvt. Ltd.-New Delhi.

<Course Code>					Modern Physics Laboratory (For CH, PE, BT)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- Understand the characteristics of waves, semiconductor, heat pump, LASER.
- Enhance knowledge on application of physics in engineering
- To develop intellectual communication skills and discuss the basic understanding of various experimental principles involved
- Demonstrate practical knowledge by applying experimental methods to correlate with the theory.
- Apply the analytical techniques and graphical analysis to the experimental data

LIST OF EXPERIMENTS

1 Determining Plank's constant and inverse square law
2 Study of Photoconductivity
3 Study of Bio-Savart's Law
4 Determining e/m by Thomson's method
5 Study of Hall Effect.
6 Experiments on single and double slit diffraction and interference with He-Ne Laser
7 Study of I-V characteristics of p-n diode.
8 Determination of thermal conductivity of different solids
9 To measure resistivity of semiconductor by Four Probe method and determination of band gap.
10 Study of Interference using Newton's Ring experiment.
11 To study G.M. tube characteristics and to calculate the dead time.
12 Energy calibration of CsI:Tl radiation detector and energy analysis of an unknown gamma source
13 To determine the numerical aperture of a given fibre optics cable using the far field measurements. 14 Experiments with heat pump
15 Study of Polarization of light using LASER

COURSE OUTCOMES

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

CO1 : Analysis the engineering problems and design the components for the solution
CO2 : Developing skills to utilize the different tools for engineering problems
CO3 : Analyse the results and correlate with theory and its application in industries
CO4 : Design the set-up and utilize for component analysis
CO5 : Identifying the problem and creating the solutions for research and development
CO6 : Analyse the scientific data and learn to be efficient as individual and a team member

TEXT/REFERENCE BOOKS

1. W.R. Runyan , Semiconductor Measurements and Instrumentation, McGraw Hill.
2. Sayer M. & Mansingh A., Measurement, Instrumentation & Experiment Design in Physics and Engineering, Prentice Hall India.
3. Melissinos A.C. and Napolitano J, Experiments in Modern Physics, Academic Press.
4. Nakra B.C. &. Chaudhery K.K , Instrumentation Measurements & Analysis, Tata McGraw Hill.
5. ORTEC Lab Manual, Experiments in Nuclear Science, ORTEC.

<Course Code>					Engineering Physics (For ME, Civil)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To master the concepts of vector algebra and the fundamentals of mechanics.
- To explore the principles of elasticity in solids, band theory of solids and the Physics of semiconductors.
- To investigate the behavior of waves and oscillations, along with their applications in acoustics and ultrasonics.
- To gain insights into thermal physics, covering conduction, convection, and radiation, and their practical applications.

UNIT I: MECHANICS **12 Hrs.**

Introduction to vector algebra, Vector fields and scalar fields, Concepts of gradient, Divergence and Curl, Gauss-Divergence Theorem, Stokes' theorem and its applications, The everyday forces of Physics, Work-energy theorem, Rigid body motion and the conservation of angular momentum, Center of mass, Center of gravity, Moment of inertia: Definition, Moment of inertia of simple planar laminas.

UNIT II: PHYSICS OF SOLIDS **10 Hrs.**

Elasticity, Stress & Strain, Hook's law, Young's modulus, Poisson's ratio, Rigidity modulus, Relationship between elastic constants, Properties of solids, electrical conductivity, Wiede-mann Franz Law, Band structure of conductor, insulator and semiconductor, Intrinsic and extrinsic semiconductor, Conductivity of semiconductors, charge concentrations, Hall effect, LED and Solar cells, Magnetism and its origin, Magnetic properties of solids, Physics of superconductors, Applications.

UNIT III: WAVES, OSCILLATIONS AND ACOUSTICS **08 Hrs.**

Waves and oscillations: Types of waves, Simple harmonic motion, Damped simple harmonic motion, types of damping, Forced oscillation, Resonance, Energy Transport in Wave motion. Acoustics & Ultrasonic: Introduction to Sound, Sabine's reverberation theory, Acoustical defects and their remedies, Doppler Effect, Ultrasonic waves, Methods of their generation and detection, Properties and applications of ultrasonic waves.

UNIT IV: THERMAL PHYSICS **12 Hrs.**

Thermodynamic systems (closed and open), Thermodynamic properties and equilibrium, Concepts of heat and work, Laws of thermodynamics, Carnot cycle, Heat transfer-thermal expansion of solids and liquids, Conduction in solids, Thermal conductivity, Forbe's method, Lees' disc method, Thermal insulation and its applications, Thermal Convection and its applications, Newton's law of cooling, Thermal Radiation – emission and absorption radiation, emissive power, Black body radiation, Stefan's laws, Wien's law.

HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : Describe fundamental concepts of mechanics, such as vector algebra, everyday forces of Physics, Moment of inertia, etc.
CO2 : Explain the principles underlying the physics of solids, including elasticity and the behavior of energy bands.
CO3 : Solve problems involving mechanics, solids, waves, oscillations, and acoustics.
CO4 : Analyze the behavior of waves, oscillations, and thermal phenomena, including their practical applications.
CO5 : Critically evaluate the significance of thermodynamic laws and superconductivity theories in solving real-world problems.
CO6 : Develop solutions and innovative applications in mechanics, solids, waves, and thermal physics by integrating concepts and principles learned throughout the course.

TEXT/REFERENCE BOOKS

1. D. Kleppner and R. J. Kolenkow, An Introduction to Mechanics, Tata McGraw-Hill.
2. D. Halliday, R. Resnick, J. Walker, Fundamentals of Physics, John Wiley & Sons.
3. Sears and Zemansky, University Physics, Pearson publications.
4. M. N. Avadhanulu, P.G. Kshirsagar, A textbook of Engineering Physics, S Chand & Co. Ltd.
5. F. M. A. Wahab, Solid State Physics: Structure and Properties of Materials, Narosa Publishing House Pvt. Ltd.
6. Richard, Oscillations and Waves: An Introduction, Taylor & Francis.
7. Y. A. Cengel, Introduction to Thermodynamics and Heat Transfer, Tata McGraw-Hill.
8. B. Lal and N. Subramaniyam, Heat and Thermodynamics, S Chand & Co. Ltd.

<Course Code>					ENVIRONMENTAL SCIENCE					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	I A	LW	LE/Viva	
2	0	0	2	2	25	50	25	--	--	100

COURSE OBJECTIVES

- To develop a comprehensive perspective of environment and sustainable development
- To understand the causes and effects of various types of pollution
- To develop an understanding of the various strategies for controlling the pollution
- To introduce the emerging environmental domains

UNIT I : INTRODUCTION TO ENVIRONMENT **05 Hrs.**

Sustainable Development; Sustainable Development Goals; Environmental Studies – Its importance and Multidisciplinary nature, Introduction to Environmental Parameters and their standards (air, water, soil, noise, etc.); Ecosystem and its types, Ideal ecosystem, Biodiversity : Its importance and conservation.

UNIT II : MULTI-SCALE ENVIRONMENTAL POLLUTION (GLOBAL, REGIONAL AND LOCAL) **06 Hrs.**

Pollution, Causes and Effects of different types of pollution : Air Pollution, Water Pollution, Soil Pollution, Solid Waste (organic and Inorganic) Pollution, Hazardous Waste Pollution, Marine Pollution, Noise Pollution, Thermal Pollution, Radioactive Pollution; Introduction to man-made disasters like floods, heat waves, landslides, etc., Introduction to the various instruments for measuring air pollution, water pollution, noise, etc.

UNIT III : ENVIRONMENTAL POLLUTION CONTROL STRATEGIES **09 Hrs.**

Multi-approaches for reducing various types of pollution: Introduction to Water and Wastewater treatment technologies, Air and Noise pollution control techniques, Introduction to different environmental management concepts like Swachh Bharat Mission, Mission LiFE (Lifestyle For Environment), etc. Indian Culture and Traditional Wisdom for managing environment

UNIT IV: EMERGING ENVIRONMENTAL MANAGEMENTS DOMAINS **08 Hrs.**

Concept of Zero Liquid Discharge (ZLD) and the reuse of the treated wastewater, Green Credit Rules - 2023, Clean Development Mechanisms (CDM) and Carbon Credits, Green Buildings, Carbon Footprint and Water Footprint, Green Business, International Environmental Laws, Environmental Auditing

HOURS: 28 Hrs.

COURSE OUTCOMES:

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

CO-1: Demonstrate comprehension of sustainable development and environmental aspects.
CO-2: Recognize the interdisciplinary characteristics inherent in Environmental studies.
CO-3: Evaluate the impact of various pollutants on the environment.
CO-4: Assess the efficacy of different technologies for environmental pollution control.
CO-5: Analyze different environmental management policies and their implications.
CO-6: Synthesize knowledge about emerging environmental management paradigms.

TEXT-BOOK AND REFERENCE BOOKS:

1. Bharucha Erach, Textbook for Environmental Studies, UGC New Delhi.
2. Daniel B. Botkin & Edwards A. Keller, Environmental Science, Wiley India edition.
3. Miller T. G. Jr., 2006. Environmental Science, Clengage Learning.
4. R. Rajagopalan, Environmental Studies, Oxford University Press.
5. Gilbert Masters and Wendell P. Ela, Introduction to Environmental Engineering and Science, PHI.

<Course Code>					Workshop Practice					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To understand safety in various manufacturing processes.
- Learn how to use various measuring tools for engineering applications.
- Hands on training and preparation of job using wood, metal and sheet as per drawing.
- Understand various manufacturing processes like machining, welding, soldering and 3D printing for prototypes.

LIST OF EXPERIMENTS

- Introduction to Workshop safety, layout and identification of various materials- plastic, wood, metals-ferrous and nonferrous, rubber, glass etc.
- Use of measuring tools for engineering applications
- Fitting job: Detailed drawing of work piece, use of fitting tools and job preparation.
- Carpentry job: Detailed drawing of work piece, use of carpentry tools and job preparation.
- Sheet metal job: Detailed drawing of work piece, use of sheet metal working tools and job preparation.
- Plumbing job: Internal/External threading, piping network using Tees, Elbows, Reducer, Bends etc
- Hands on training on mini lathe and milling machine
- Demonstration of welding, brazing and soldering
- Soldering and desoldering for PCB
- 3D printing using polymer and metal.

COURSE OUTCOMES

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

CO1 : Define workshop safety and various engineering materials
CO2 : Understand various measuring equipment
CO3 : Apply various workshop tools in preparing job for carpentry, fitting, sheet metal and plumbing
CO4 : Examine various manufacturing operations like welding and machining
CO5 : Evaluate soldering operation for PCB
CO6 : Create prototype using 3D printing

TEXT/REFERENCE BOOKS

1. S. K. Hajra Choudhury, Elements of Workshop Technology, Vol. I & II, Media Promoters and Publishers.
2. H. S. Bawa, Workshop Practice, Tata-McGraw Hill.
3. Kalpakjian S. And Steven S. Schmid, Manufacturing Engineering and Technology, Pearson Education India Edition.
4. Rao P.N., Manufacturing Technology, Vol. I and Vol. II, Tata McGraw Hill House.

<Course Code>					Biology For Engineers					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
2	0	0	2	2	25	50	25	--	--	100

COURSE OBJECTIVES

- To understand the basic biological concepts and their engineering applications.
- To introduce the students with an understanding of biodesign principles to create novel devices and structures.
- To provide the students an appreciation of how biological systems can be re-designed as substitute products for natural systems.
- To study the development of interdisciplinary vision of biological engineering.

UNIT I: INTRODUCTION TO BIOMOLECULES **7 Hrs.**

Cell as a basic unit of life, prokaryotic and eukaryotic cells, microbes, plant and animal cells; Cell organelles – structure and function; Cell membrane. Vitamins and Minerals, Carbohydrates, Lipids, Amino Acids and proteins, Nucleic Acids (DNA and RNA).

UNIT II: NATURE-BIOINSPIRED MECHANISMS **7 Hrs.**

Brain as a CPU system (architecture, CNS and Peripheral Nervous System, signal transmission, EEG, Robotic arms for prosthetics). The eye functions as a camera. Heart as a pump system, The ear and nose as signal transmission models. Lungs operate as an exchange model, kidneys function as a filtration system. Echolocation (ultrasonography, sonars), Photosynthesis (photovoltaic cells, bionic leaf). Bird flying (GPS and aircrafts), Lotus leaf effect (Super hydrophobic and self-cleaning surfaces), Plant burrs (Velcro).

UNIT III: TRENDS IN BIOENGINEERING **7 Hrs.**

Stem cell and its applications, Bio printing techniques and materials. Applications of Bioinformatics, Artificial Intelligence for diagnosis and detection of communicable and non-communicable diseases, Biosensors in healthcare, Environmental monitoring, Food Safety and Biosecurity.

UNIT IV: ENGINEERING PERSPECTIVES OF BIOLOGICAL SCIENCES **7 Hrs.**

Biology and engineering crosstalk – At cell level: Hybridoma technology, At tissue level: Plant Tissue Culture, Animal Tissue Culture; Tissue Engineering, Introduction to Biomimetics and Biomimicry, Nanobiotechnology, Introduction to Radiology, High-throughput diagnostics in clinics: Molecular Diagnostics (PCR), DNA chips.

HOURS: 28 Hrs.

COURSE OUTCOMES

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

CO1 : Gain fundamental knowledge of origin of life.
CO2 : Demonstrate the involvement of major organs and systems in the human body as models for bioengineering design.
CO3 : Analyse the scope and opportunities of Implementing bioengineering concepts relates to advanced imaging and diagnostic techniques for clinical diagnosis and disease management.
CO4 : Get acquainted with concepts of bioengineering trends including biosensors, AI , Imaging techniques in disease diagnosis and treatment
CO5 : Focus on implications of Bioinspired designs, Bioengineering trends including, tissue engineering and molecular diagnostics.
CO6 : Think critically towards exploring innovative biobased solutions for socially relevant problems.

TEXT/REFERENCE BOOKS

1. Thyagarajan S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W., Barathi S., and Jaganthan M., Biology for Engineers, Tata McGraw-Hill, New Delhi.
2. Arthur T. Johnson, Biology for Engineers, CRC Press, Taylor and Francis.
3. Geoffrey M.Cooper, The Cell: A molecular Approach, ASM Press.
4. Sohini Singh and Tanu Allen, Biology for Engineers", Vayu Education of India, New Delhi.
5. Yoseph Bar-Cohen, "Biomimetics: Nature-Based Innovation, CRC Press.
6. Stuart Fox, Krista Rompolski, Human Physiology", McGraw-Hill (eBook).
7. D. Floreano and C. Mattiussi, Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies, MIT Press.

UG_1_T					Computer Programming-I					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
1	0	0	1	1	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the usage of operators and data types.
- To apply different types of Conditional and looping statement.
- To create different types of data collections.
- To implement user defined function.
- To perform different operations upon files.

1: Basics of Programming **3 Hrs.**

Introduction to Computer Programming, Features of C language, Structure of C program, program execution flow, C Tokens, variables, Data types, Operators, Decision control statements-if, switch, go to statement. Loop control structures- while, do-while, for loop, Break statement, Continue statement

UNIT 2: Derived Data types **4 Hrs.**

Array: one dimensional and multidimensional array, Declaration, initialization, Array Manipulations. Matrix operations, String-Basic Concepts, Inbuilt String manipulation Functions, Pointer, Pointer arithmetic, Pointer to pointer, Array of Pointers

3: Function and Structure **4 Hrs.**

Introduction to user defined functions, Types of Functions, Call by value-call by reference, recursion, pointers to functions, Structures, Array of Structure, Union

4: Files Handling **3 Hrs.**

File handling in C, Different types of files, Operations on Files such as File creation, File deletion, File access modes such as read, write, append, File concatenation, File handling using seek function.

14 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1: Understand basics of programming.
CO2: Identify different programming constructs for a problem.
CO3: Apply appropriate derived data type for data storage.
CO4: Prepare a user defined data type based on data attributes.
CO5: Construct user defined functions for problem solving.
CO6: Analyse different data structure based on application requirement.

TEXT/REFERENCE BOOKS

1. Kernighan & Ritchie, C Programming Language‟, PHI
2. K. N. King , "C Programming: A Modern Approach" , W.W. Nortorn
3. David Griffiths and Dawn Griffiths, "Head First C: A Brain-Friendly Guide", O'Reilly
4. E.Balaguruswamy, Programming in ANSI C, McGraw-Hill
5. Y.P. Kanetkar ,"Let us C", BPB Publication
6. Y.P. Kanetkar , "Pointers in C", BPB Publications

UG_1_P					Computer Programming – I Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To implement basic programming concepts.
- To create different types of data collections.
- To implement user defined function.
- To perform different file handling operations.

List of Experiments:

1. **Introduction to Computer Programming**:
   Understanding compilation process through a simple C program, program execution flow, C Tokens, variables and keywords and identifiers, types of C constants and variables.
2. **Simple and formatted Input Output Operations:**
   Data types, Operators, Input /output statements in C, Formatted I/O, format specifiers, escaper sequences
3. **Decision making and branching:**
   if, if-else, if-else ladder, switch, go to statement, conditional operator statement
4. **Looping control structures:**
   while, do-while, for loop, Break statement, Continue statement
5. **Derived Data Type: Array and Strings:**
   One dimensional and multidimensional array, Declaration, initialization, Array Manipulations. Matrix operations, Basic Concepts, Inbuilt String manipulation with and without using inbuilt functions.
6. **Derived Data Type: Structure and Union:**
   structure, arrays and structures, structures and functions, pointer to structure, typedef, unions
7. **Functions:**
   Introduction to user defined functions, Types of Functions, Call by value-call by reference, header file creation, recursion, pointers to functions, arrays and functions
8. **Pointers:**
   Pointer's basics, use of &, * operator in context to pointers, Pointer arithmetic, Array and String processing using pointer, pointer to pointer, Array of Pointers
9. **File Handling in C:**
   File handling in C, Different types of files, Operations on Files, File handling functions.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1: Identify the use of appropriate naming conventions and programming style including appropriate comment density.
CO2: Implement a basic C program using appropriate control structure.
CO3: Apply appropriate derived data types based on data attributes.
CO4: Develop solutions with pointers and utilize them to access strings and structures.
CO5: Design user defined functions for problem solving and reuse them across different programs.
CO6: Apply suitable file handling functions and operations.

TEXT/REFERENCE BOOKS

1. Kernighan & Ritchie, C Programming Language", PHI
2. K. N. King , "C Programming: A Modern Approach", W.W. Norton
3. E. Balaguruswamy, Programming in ANSI C, McGraw-Hill
4. Y.P. Kanetkar,"Let us C", BPB Publication
5. Y.P. Kanetkar, "Pointers in C", BPB Publication

<Course Code>					Universal Human Values					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
1	0	0	1	1	25	50	25	--	--	100

COURSE OBJECTIVES

- To understand the need of nurturing human values through the process in value-based education system.
- To understand and develop a holistic perspective on self-exploration and being in harmony with family, society and nature.
- To facilitate the students in understanding harmony at all the levels and applying in their profession and work place to lead an ethical life.

UNIT I: HUMAN VALUES AND PROCESS OF VALUE EDUCATION 04 Hrs.

Human values, human aspirations and the ultimate goal, understanding happiness and prosperity, appraise the meaning of satisfaction and happiness in current scenario, harmony and compatibility, values imbibed education system and process

UNIT II: KNOWING SELF - HARMONY WITH SELF 04 Hrs.

Understanding self, capabilities and challenges, understanding material (physical facilities) and spiritual needs - need of mind and body, understanding body as an instrument, harmony between mind and body, synchronizing physical health and mental health, practicing healthy habits for healthier me

UNIT III: HARMONY IN RELATIONSHIP – FAMILY, SOCIETY AND NATURE 03 Hrs.

Harmony in relationships, values for harmony in any human-human interaction, harmony in family, and society, trust and respect for others, self-esteem and ego, equality, equity, inclusion and liberation, concept of '*Vasudhaiva Kutumbakam',* understanding co-existence and sync with nature

UNIT IV: HARMONY IN PROFESSION AND ETHICAL BEHAVIOR 03 Hrs.

Ethical human conduct, acceptance and respect, appraising the qualities of others, professional competence for enabling harmony in system and enabling universal human order, scope of eco-friendly systems, strategies to reach the harmonious ecosystem to reach Universal Human Order '*Sarvabhauma Vyavastha'*

TOTAL HOURS: 14 Hrs.

COURSE OUTCOMES

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

CO1 - Understand the significance of human values, its need, and process of value education.

CO2 - Appraise the meaning of happiness and prosperity as short- and long-term goal of life. Understand them and in context of the current scenario

CO3 - Distinguish between the mind and body, physical and spiritual wellbeing for harmony within self

CO4 - relationships to build harmonious society

CO5 - Understand the importance of harmony with nature and appreciate co-existence for harmonious ecosystem.

CO6 - Create the perfect professional place and work environment following the ethical practices and strategize to uphold the human values at all the levels and interactions.

TEXT/REFERENCE BOOKS

1. R. R. Gaur, R Sangal, G P Bagaria, A foundation course in Human Values and Professional Ethics, Excel books.
2. A. Nagraj, Jeevan Vidya ek Parichay, Divya Path Sansthan, Amarkantak.
3. A. N. Tripathi, Human Values, New Age Intl. Publishers.
4. M. K. Gandhi. The Story of My Experiments with Truth, Fingerprint Publishing.
5. Ivan Illich, Energy & Equity, The Trinity Press, Worcester, and Harper Collins.
6. E. F. Schumacher, Small is Beautiful: a study of economics as if people mattered, Blond & Briggs, Britain.
7. Sussan George, How the Other Half Dies, Penguin Press.

2nd Semester

<Course Code>					Mathematics – II					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs. / Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	--	--	100

COURSE OBJECTIVES

- Understand and apply principles of complex differentiation and integration effectively.
- Solve systems, find eigenvalues, and apply transformations confidently.
- Solve various ODEs, apply methods, and tackle engineering problems.
- Analyze periodic functions, derive series, and apply in diverse applications.

UNIT 1 COMPLEX DIFFERENTIATION AND INTEGRATION **12 Hrs.**

Limit, Continuity, Differentiability of the function of a complex variable, Analytic function, Cauchy-Riemann equation (in Cartesian coordinates), Harmonic function and its significance, Singularities, Definition of a Complex line integral, Contour integrals, Cauchy integral theorem, Cauchy Integral formula (CIF), CIF for derivatives, Taylor's series and Laurent Series , Calculation of residues, Cauchy Residue theorem, Applications of residues to evaluate real definite integrals.

UNIT 2 MATRIX ALGEBRA AND ITS APPLICATIONS **10 Hrs.**

Solution of system of algebraic equation - Rank of a matrix, consistency of system of equation - Characteristic equation of a square matrix- Eigen values and Eigenvectors of a real matrix - Properties of Eigen values and Eigen vectors - Cayley-Hamilton theorem (without proof) - finding inverse of a matrix - Diagonalization of a matrix.

UNIT 3 ORDINARY DIFFERENTIAL EQUATIONS WITH APPLICATIONS **10 Hrs.**

Differential equations of Higher-order differential equations with constant coefficients, Rules for finding C.F. and P.I., Method of variation of parameters, Cauchy and Legendre's linear equations, Linear differential equations of second order with variable coefficients; Simultaneous linear equations with constant coefficients, applications of higher-order differential equations in solving engineering problems.

UNIT 4 FOURIER SERIES **10 Hrs.**

Periodic functions, Odd and even functions, Euler's formulae for Fourier series in an interval of length 2 pi, Change of interval, Dirichlet's conditions, Half range Sine and Cosine series, Complex Fourier series, Parseval's identity and its applications.

HOURS: 42 Hrs.

COURSE OUTCOMES

On completion of the course, students will be able to

CO1 : Recall fundamental concepts of complex differentiation, matrix algebra, differential equations, and Fourier series.

CO2 : Understand the significance of complex differentiation, integration, Matrix, ODE and Fourier series in respective contexts.

CO3 : Apply the concept of complex function, Matrix, ODE and Fourier series to extract the solutions of engineering problems

CO4 : Analyze the use of complex variable Matrix, ODE and Fourier series in engineering problems.

CO5 : Assess the significance and effectiveness of mathematical concepts and theorems in solving real-world problems, particularly in engineering and scientific applications.

CO6 : Design and construct solutions to complex mathematical problems using a variety of techniques, including transformations, mappings, and advanced methods in differential equations.

TEXT/REFERENCE BOOKS

1. R.V. Churchill and J. W. Brown, Complex variables and Applications, McGraw-Hill.
2. J. M. Howie, Complex Analysis, Springer-Verlag.
3. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, Alpha Science.
4. Erwin Kreyszig, Advanced Engineering Mathematics", John Wiley.
5. G. Strang, Linear Algebra and its Applications, Cengage Learning.
6. K. Hoffman and R. A. Kunze, Linear Algebra", Prentice Hall of India.

<Course Code>					Discrete Mathematical Structures (For CS, ICT)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	--	--	100

COURSE OBJECTIVES

- To understand the concept of sets, relations, functions and logic.
- To understand Combinatorics as an analytical method for problem solving.
- To apply graph theory-based modeling to solve real life problems.
- To explain the basics of algebraic structures.

UNIT I: SETS, RELATIONS, FUNCTIONS AND LOGIC **10 Hrs.**

Finite and Infinite sets, Counting Principle, Principle of inclusion and exclusion, Mathematical Induction, Relations, Types of Relation, Equivalence relation, Partial Ordered Relations. Functions, types of functions, Logic and Propositional calculus – Simple and Compound Propositions, Basic logical operators. Truth tables. Tautologies, Contradictions, logical equivalence, conditional and bi-conditional statements. Valid arguments and Fallacy. Hasse diagram and Lattice.

UNIT II: COMBINATORICS **10 Hrs.**

Permutation, Combination, Pigeonhole principle, Extended Pigeonhole principle, Recurrence relations, Linear recurrence relations with constant coefficients, Solutions of recurrence relations: second order homogeneous linear recurrence relation, general homogeneous linear recurrence relation.

UNIT III: GRAPH THEORY AND ITS APPLICATIONS **12 Hrs.**

Graphs and related definitions, Sub graphs, Homomorphism and Isomorphism, Paths and Connectivity. Traversable and Eulerian graphs, and Konigsberg Bridge problem, Hamiltonian graphs, Labeled and weighted graphs. Complete, regular and bipartite graphs, Planar graphs, Graph coloring. Four color problem. Directed graphs, Strongly and weakly connected graphs, Trees, rooted trees, sequential representation of directed graphs, Adjacency matrix, Powers of the adjacency matrix.

UNIT IV: ALGEBRAIC STRUCTURES **10 Hrs.**

Group, Semi group, Monoids, Properties of a Group, Composition table for finite Group, Order of a group, Order of its elements, Cyclic Group, Generator, Lagrange's Theorem. Ring, Properties of Rings, Integral Domain, Field.

TOTAL HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : Remember the basics of counting and combinatorics.
CO2 : Understand the basic concepts of sets, relations, functions, logic and be able to determine their properties.
CO3 : Model and solve the real life problem using recurrence relations.
CO4 : Apply Graph theory in related areas such as shortest path problems and network flow problems.
CO5 : Defend and point out fallacious reasoning and propositions in algebraic structures.
CO6 : Model and solve any given engineering problem involving graphs and trees.

TEXT/REFERENCE BOOKS

1. Seymour Lipschutz and Marc Lipson, "Discrete Mathematics, Schaum's Series", McGraw-Hill Education.
2. Kenneth Rosen, "Discrete Mathematics and Its Applications", McGraw Hill Education.
3. Bernard Kolman, Robert Busyb and Sharon C. Ross, "Discrete Mathematical Structures", Pearson.
4. Thomas Koshy, "Discrete Mathematics with Applications", Academic Press Inc.
5. Ralph P. Gramaldi, "Discrete and Combinatorial Mathematics", Pearson.
6. C.L. Liu, D.P. Mohapatra, "Elements of Discrete Mathematics: A Computer Oriented Approach", McGraw Hill Education.

<Course Code>					Engineering Chemistry					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To develop the fundamental understanding about traditional materials.
- To provide the knowledge about structural features, synthesis, properties of various categories of advanced materials.
- To develop the knowledge and skills for different characterization techniques of materials.
- To provide the knowledge about the role of chemistry in modern engineering applications.

UNIT I: CHEMISTRY OF ENGINEERING MATERIALS **12 Hrs.**

Traditional Materials: Introduction and classification of materials; metallic materials, polymeric, ceramic materials Advanced Materials: Introduction to nanomaterials: Properties and application; Carbonaceous materials (fullerene, carbon nanotube, graphene, etc.); Composite materials; Liquid crystals: Classification and Application

UNIT II: MODERN ANALYTICAL TECHNIQUES **10 Hrs.**

Instrumentation, principle and characterization of materials: X-ray diffraction (XRD), Electro analytical techniques; FTIR, UV-visible spectroscopy; Thermal analysis (TGA-DTA-DSC); Chromatographic techniques (GC, HPLC)

UNIT III: ADSORPTION, CATALYSIS AND KINETICS **10 Hrs.**

Adsorption - Characteristics, Classification, Application , Adsorption isotherms- Freundlich, Langmuir & BET Chemical Kinetics - Rate law, Arrhenius equation, Transition state theory, Collision theory; Complex reactions Catalysis - Homogeneous and Heterogeneous Catalysis; Mechanism of Catalysis; Industrial Applications of catalysts

UNIT IV: CHEMISTRY OF ENERGY DEVICES **10 Hrs.**

Principles of primary and secondary batteries, Fuel Cells and their operation principles, Principles and uses of supercapacitors; Photocatalytic hydrogen production: Principles and challenges; Traditional and new generation solar cells.

TOTAL HOURS: 42 Hrs.

COURSE OUTCOMES

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

CO1 : **Remember** the structural features and properties of different classes of traditional materials

CO2 : **Classify** advanced materials like nanomaterials, carbonaceous and composite materials, and liquid crystals

CO3 : **Apply** the skills by understanding various instrumental techniques for characterisation of materials.

CO4 : **Analyze** the key concepts in engineering chemistry viz. adsorption and chemical kinetics and laterally ponder over the applications of such concepts in engineering challenges.

CO5 : **Justify** the important insights into the industrial application of different types of catalysis via analysing mechanisms of catalysis.

CO6 : **Develop** the knowledge on the role of chemistry in various modern engineering applications such as in energy devices.

TEXT/REFERENCE BOOKS

1. W.D. Callister, **"An Introduction to Materials Science & Engineering",** John Wiley & Sons (2007).
2. MW Barsoum, **"Fundamental of Ceramics",** IOP publishing (2003).
3. T. Pradeep, "Text book of Nanoscience and Nanotechnology", Mc. Graw Hill Education (2003).
4. Murty, Shankar, B Raj, Rath, Murday, **"Textbook of Nanoscience and Nanotechnology",** Springer (2013).
5. V. Raghavan, **"Materials Science and Engineering",** Prentice-Hall of India Private Limited (2003).
6. A. Douglas, Donald Skoog, M.West**, "Principles of Instrumental Analysis",** Cengage, 6th Edition, (2014)
7. Jain & Jain, **"Engineering Chemistry"** by Dhanpat Rai Publishing Company

20CH101P					Engineering Chemistry Practical					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To enhance and develop scientific and analytical skills
- To relate concepts learned in chemistry and engineering to the real-world situations.
- To acquire skills to perform laboratory experiments.
- To demonstrate safe and proper use of standard chemistry glassware and equipment.

LIST OF EXPERIMENTS

1. **Iodometry**– To determine the strength of given copper sulphate solution by titrating against N/20 sodium thiosulphate (hypo) solution
2. **Iodimetry**– To determine the strength of given ascorbic acid by titrating against standard N/10 iodine solution
3. **Complexometric Titration**– To determine the total, permanent and temporary hardness of given water by complexometric titration using standard 0.01M EDTA solution
4. **pH metric titration**– To determine the strength of given HCl solution using a standard NaOH solution by performing a pH-metric titration
5. **Conductometric titration**– To determine the strength of given HCl solution using a standard NaOH solution by performing a conductometric titration
6. **Chemical Kinetics**– To study the kinetics of decomposition of sodium thiosulphate by a mineral acid
7. **Drawing chemical structures** - To Draw Chemical Structures of organic molecules using ChemDraw
8. **Colorimetric determination**: To determine the concentration of copper present in the effluent of electroplating industries by using colorimeter.
9. **Detection of biomolecule**: Detection of the presence of carbohydrates in test solution by using Benedict's reagent
    - **Preparation of drug molecule**: Preparation of Aspirin from salicylic acid
- **Polymerization**– To prepare a polymer (Nylon 6,10), identify the functional groups by FT-IR

COURSE OUTCOMES

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

CO1 : **Recall** the concepts learned in chemistry and engineering to the real-world situations.
CO2 : **Show** the ability to identify, analyze and interpret the results from the experiments
CO3 : **Experiment** with the instrumental method using conductometer and pH meter
CO4 : **Analyze** compounds by titrimetric, gravimetric and instrumental methods
CO5 : **Determine** the concentration of unknown solutions by spectrophotometric method
CO6 : **Predict** the reaction rate and predict the order and rate constant

TEXT/REFERENCE BOOKS

1. V. K. Ahluwalia, S Dhingra, A Gulati, **"College Practical Chemistry",** Universities Press
2. J.B. Baruah, P Gogoi, **"Foundations of Experimental Chemistry",** PharmaMed Press.
3. S.S. Sawhney, M.S. Jassal, S.P. Mittal, "**A Text Book of Chemistry Practical's"** Vol I & II, APH Publishing Corp.

<Course Code>					Elements of Electrical and Electronics Engineering					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To impart knowledge on DC and AC circuits.
- To learn construction, working principle and characteristics of transformer and induction machines.
- To introduce students to various means for electrical safety, protection of electrical installations and Batteries.
- To understand rectification through p-n junction diode, applications of diode and Transistor Characteristics.

UNIT I: DC CIRCUITS **10 Hrs.**

Electrical circuit elements (R, L and C), voltage and current sources, dependent and independent sources, Ohms Law, temperature co-efficient of resistance, Kirchhoff current and voltage laws, voltage and current divider circuit, Thevenin and Norton Theorems and their equivalents, maximum power transfer and superposition theorems, nodal and mesh analysis, star-delta transformation, Time domain analysis/natural response of first order RL and RC Circuit

UNIT II: AC CIRCUITS **10 Hrs.**

Generation of AC voltage, representation of sinusoidal waveforms, RMS values of different sinusoidal waveforms, Rectangular and Polar representation of phasor, Sinusoid representation in time and frequency domain, Analysis of single-phase ac series circuits consisting of R, L, C, RL, RC, RLC combinations, instantaneous power, average power and reactive power, complex power and power factor.AC parallel circuit and its solution in admittance form, resonance in AC series and parallel circuits, polyphase circuits, star and delta representation of polyphase circuits, power measurement in polyphase circuits.

UNIT III: INTRODUCTION TO ELECTRICAL MACHINES AND ELECTRICAL INSTALLATION **12 Hrs.**

Magnetic materials and its B-H characteristic, Faraday's Law of Electromagnetic Induction; **Single Phase Transformers:** Construction, working principle, types, EMF equation, ideal transformer, practical transformer, Operation of transformer on no-load and load, **Induction Machine:** construction, types of 3-phase induction motors, working principle, production of rotating magnetic field, operation, starting and running torques, Torque-slip characteristics, Power Stages in an induction motor.

Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing, Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, Electrical safety rules.

UNIT IV: BASIC OF ELECTRONICS AND ELECTRICAL INSTALLATION **10 Hrs.**

Semiconductor Diodes and Applications, Diode as a Switch, Diode as a Rectifier, Half Wave and Full Wave Rectifiers with and without Filters; Breakdown Mechanisms, Zener Diode – Operation and Applications; Opto-Electronic Devices–LEDs, Photo Diode and Applications; Silicon Controlled Rectifier (SCR) – Operation, Construction, Characteristics, Ratings, Applications.

Transistor Characteristics covering, Bipolar Junction Transistor (BJT) – Construction, Operation, Amplifying Action, Common Base, Common Emitter and Common Collector Configurations, Field Effect Transistor (FET) – Construction, Characteristics of Junction FET.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 – Understand electrical circuits using network theorems.

CO2 – Compare the behaviour of R, L and C and their combinations in AC circuits.

CO3 – Analyze balanced polyphase systems in star and delta configurations.

CO4 – Understand the construction, working and basic characteristics of transformer and induction machines.

CO5 – Recognize the importance of protective devices, electrical safety measures and characteristics of Batteries.

CO6 -- Demonstrate application of different diodes in circuits and Understand the Transistor characteristics.

TEXT/REFERENCE BOOKS

1 J. Bird, "Electrical Circuit Theory and Technology", Routledge, Tailor and Francis Group, Sixth Edition.

2 D. P. Kothari and I. J. Nagrath, "Basic Electrical Engineering", Tata McGraw Hill.

3 B. L. Theraja, "Electrical Technology", Vol. 1, S. Chand Publication, New Delhi.

4 Surjit Singh, "Electrical Estimating and Costing", Dhanpat Rai and Co.

5 Boylestad and Nashlesky, "Electronic Devices and Circuit Theory", PHI.

24EE101P					Elements of Electrical and Electronics Engineering Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To gain practical knowledge on DC and AC circuits
- To learn operation of electrical instruments and electrical machines
- To introduce students to various means for electrical safety, protection of electrical installations
- To understand characteristics and applications of semiconductor diodes and transistors.

LIST OF EXPERIMENTS

1. Introduction to elements of electrical engineering laboratory and to study different electrical measuring instruments.
2. To Verify Ohm's law with linear resistors and find power dissipation in resistor.
3. To validate Thevenin and Norton theorem for DC circuits.
4. To validate Superposition and Maximum Power Transfer theorem for DC circuits.
5. To evaluate the AC R, L and R-L series circuit performance and to measure the active power, reactive power and apparent power connected with single phase AC supply.
6. To evaluate performance of AC R-L parallel circuit and to measure the active power, reactive power and the apparent power connected with single phase AC supply.
7. To perform a direct load test on a single-phase transformer and determine the efficiency and voltage regulation at different loads.
8. To measure three phase power using two wattmeter method.
9. To draw the time-current characteristics of MCB and study the function of ELCB.
10. To obtain the VI characteristic of silicon and germanium diodes.
11. To obtain reverse characteristics of Zener diode.
12. To study half wave, full wave and bridge rectifiers.
13. To determine the DC Characteristics of BJT in CE Configuration.

COURSE OUTCOMES

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

CO1 – To understand the basic operation of electrical equipment's & measuring instruments.

CO2 – To perform various network theorems for DC circuits.

CO3 – To understand the performance of AC circuit with R, L load.

CO4 – To evaluate star and delta configuration of polyphase system and measure power in polyphase system.

CO5 – To evaluate the performance of single phase transformer.

CO6 – To understand the working principle of semiconductor diodes, transistor characteristics and its applications.

TEXT/REFERENCE BOOKS

1 J. Bird, "Electrical Circuit Theory and Technology", Routledge, Tailor and Francis Group, Sixth Edition.
2 D. P. Kothari and I. J. Nagrath, "Basic Electrical Engineering", Tata McGraw Hill.
3 B. L. Theraja, "Electrical Technology", Vol. 1, S. Chand Publication, New Delhi.
4 Surjit Singh, "Electrical Estimating and Costing", Dhanpat Rai and Co.
5 Boylestad and Nashlesky, "Electronic Devices and Circuit Theory", PHI.

<Course Code>					Engineering Graphics - Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE	
0	0	4	2	4	--	--	--	50	50	100

COURSE OBJECTIVES

- To cover the fundamental of engineering drawing and standards used in drawing.
- To explain the students to communicate ideas using orthographic and isometric projection methods.
- To help students to use CAD software to prepare drawings.
- To demonstrate the presentation of drawing using sketching and 3D modelling in CAD tool.

LIST OF EXPERIMENTS

1. Introduction to Engineering Graphics: Principles of engineering graphics and their significance, drawing instruments & accessories, lettering and numbering, types of lines, dimensioning methods, basic geometric drawing, reading a drawing.
2. Orthographic Projection: Introduction to projection, types of projection, 1^{st} angle and 3^{rd} angle projection
3. Isometric Projection: Principles of isometric projection – isometric scale, isometric views, conventions, conversion of isometric views to orthographic views and vice-versa
4. Projection of Solids and Development of Surface: Classification of solids, projections of solids like cylinder, cone, pyramid, and prism with its inclination to reference plane, development of surfaces of right regular solids - prism, pyramid, cylinder and cone.
5. Introduction of Computer Aided Engineering Drawing: Demonstrating knowledge of the theory of CAD software, use of software in drawing, CAD software user interface, commands, Coordinate System, menus and toolbars, planes, dimensioning, saving of files, Select and erase objects, zoom tools, and others
6. Basic sketching using CAD tool: Sketch entities using tools – origin, points, lines, circle, arcs, polygons, fillets and chamfer, trim, extend and offset, copy, cut, delete and others
7. Advanced sketching using CAD tool: Sketching entities using relation constrains, Mirror, Patterning, full definition of drawing and others
8. Basic 3D modelling using CAD tool: Extrude, cut, drawing on different planes, editing, symmetric, revolving, and others
9. Computer aided drawing sheets: Preparing drawing sheets, creating different views, section view, drawing templates, and others

COURSE OUTCOMES

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

CO1 : Recalling the fundamentals of engineering graphics by considering basic rules of drawing, dimensioning, and labelling.
CO2 : Explain the principle of projection using orthographic and isometric projection.
CO3 : Represent the 2-dimensional drawing using CAD tool.
CO4 : Construct the 3-dimensional geometries using CAD tool.
CO5 : Apply the concept of engineering drawing by organizing drawing views and applying necessary dimensions by preparing drawing sheets
CO6 : Analyse the intricate details of solid using projection of solid, sectioning of solid and development of lateral surfaces.

TEXT/REFERENCE BOOKS

1. Bhatt N.D., Panchal V.M. & Ingle P.R., Engineering Drawing, Charotar Publishing.
2. Shah P.J., Engineering Graphics, S. Chand Publishing.
3. Agrawal, B. & Agrawal C. M., Engineering Drawing, Tata McGraw Hill Publishers.
4. Hanifan R, Perfecting Engineering and Technical Drawing, Springer International Publishing Switzerland.
5. Corresponding Set of CAD Software Theory and User Manuals.

<Course Code>					Yoga, Health & Hygiene					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To impart the students with the basic concepts of physical education, sports, and yoga for health and wellness.
- To familiarize the students with health-related exercises, sports, and yoga for overall growth and development.
- To create a foundation for professionals in physical education, sports, and yoga.
- To impart the basic knowledge and skills to teach physical education, sports, and yoga activities.

ACTIVITY I: KINESIOLOGY AND CARDIO FITNESS TEST

Introduction to Kinesiology and the Physiological Basis of Conditioning, Sports Psychology, and the Cooer Cardio Fitness Test 12-Minutes Run/Walk: How to Start Walking/Runing: Get expert tips, tools, and training.

ACTIVITY II: YOGA

Introduction to Yogasana and Yoga Therapy: A Rehabilitation Tool and the Effect of Yoga on Exercise Endurance as Assessed by Cardiorespiratory Efficiency Tests: Studt on yogic practices that promote and improve respiratory and cardiovascular function and enhance physical fitness.

ACTIVITY III:GAMES AND SPORTS LEAGUE

Practice sessions for outdoor and indoor games, event-wise practice, and team games organized on the sport, game-wise practice as per the student's interest: Football, Chess, Cricket, Tennis, Basketball, Volleyball, Athletics (Relay), Pickleball.

COURSE OUTCOMES

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

CO1 : Discover the balance of health and happiness through the basic principles and practices of physical education, sports, and yoga.
CO2 : Understand that the students will be able to be instructed on physical activities, sports, yoga practices, theories, and rules of various games for healthy living.
CO3 : Analyze emerging trends and issues in world sports and develop leadership qualities among students to conduct, organize, and officiate physical education, sports, and yoga events at schools, colleges, and the community.
CO4 : Practice on the field and in the indoor yoga hall.
CO5 : Develop a spirit of teamwork and fair play.
CO6 : Demonstrate understanding by participating in games and sports leagues.

TEXT/REFERENCE BOOKS

1. Athletic Track and Court Marking Handbook of Games And Sports – Rajesh Agola.
2. Asana, Pranayama, and Kriyas - Swami Satyanand Swami.Munger.
3. Sports Games and Rule, Regulation - Pankaj Vinayak Pathak
4. Yogic Prakriyanche Margdarshan – Dr.M.L.Gharote - (The Lonavala Yoga Research Institute,Lonavala)

<Course Code>					National Service Scheme (NSS)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To develop a sense of civic and social responcibility.
- To identify the needs and problems of the community and involve them in problem-solving.
- To engage in creative and constructive social action.
- To develop social character and leadership through NSS.

ACTIVITY I: ENVIRONMENT AND SUSTAINABILITY -WATER - WAST MANAGEMENT

Volunteering work for Environment & Sustainability (water and waste management) and Tree Plantation.

ACTIVITY II: NSS 7 DAYS SPECIAL CAMP

Volunteering for tree planting, agriculture compost, tree guard,Gujarat Skill Development Mission, and social activities in the village as per the government NSS manual.

ACTIVITY III: FIT INDIA MISSION

Volunteering for Cardio Fitness, Yoga, Running, Mission Olympics, Self-Defense, and Agneepath Mission.

COURSE OUTCOMES

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

CO1 : Identify the needs and problems of the community.
CO2 : Understand the importance of his / her responsibilities towards society.
CO3 : Analyze the environmental and societal problems/issues.
CO4 : Evaluate the existing system and propose practical solutions for the sustainable development.
CO5 : Develop a government or self-driven projects effectively in the field.
CO6 : Understand the government or self-driven projects effectively in the society.

TEXT/REFERENCE BOOKS

1. NSS Course Manual, Published by NSS Unit,PDEU (https://www.pdpu.ac.in/nssreport.html)
2. Government of Gujarat NSS Cell (https://nss.gov.in/gujarat-1)
3. Government of India NSS Cell, Activities reports and manual (https://nss.gov.in/)

<Course Code>					National Cadet Corps (NCC)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To develop discipline, character, brotherhood, the spirit of adventure and ideals of selfless service amongst young students
- To develop youth leadership qualities in the students.
- To induce social consciousness among students through various NCC camps

ACTIVITY I: INTRODUCTION TO NCC

Introduction to NCC, aims and objectives, structure and organization of NCC, cardinals of NCC, NCC Flag, oath of NCC, NCC Song, incentives of NCC

ACTIVITY II: NATIONAL INTEGRATION

Importance of national integration and awareness, necessity, national interests, objectives, threats and opportunities, unity in diversity

ACTIVITY III: NCC CAMPS, SOCIAL SERVICE, AWARENESS AND COMMUNITY DEVELOPMENT ACTIVITIES

Social awareness & community development, health & hygiene, environment awareness and conservation, cadets will participate in various activities e.g., blood donation camp, swachhata abhiyan, constitution day, etc., participation into NCC camps like ATC, CATC, NIC, COC, TSC, RDC, leadership camps, etc.

ACTIVITY IV: DRILL, WEAPON TRAINING AND ADVENTURE ACTIVITIES

Types of drill, foot drill, general and words of command, saluting, weapon training, map reading, field craft & battle craft, Introduction to infantry weapons & equipment, obstacle and weapon training (during camps), adventure training, participation into Republic and Independence day ceremonial parades at university.

COURSE OUTCOMES

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

CO1 : Know about the history of NCC, its organization, and incentives of NCC for their career prospects.
CO2 : Understand the importance of Nation building and individual contribution to the same
CO3 : Maintain discipline and team spirit
CO4 : Build the character and leadership qualities
CO5 : Understand that drill as the foundation for discipline and to command a group for common goal.
CO6 : Develop the sense of self-less social service for better social & community life.

TEXT/REFERENCE BOOKS

1. Cadet's Handbook SD/SW- Common Subjects, all wings by DG NCC, New Delhi

UG_2_T					Computer Programming-II					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
1	0	0	1	1	25	50	25	-	-	100

COURSE OBJECTIVES

- To impart the basic concepts of Python Programming language
- To be familiar with data structures available in Python.
- To understand the concept of function & object-oriented programming.
- To use the file and exception handling for designing applications

UNIT 1 INTRODUCTION AND LANGAUGE FUNDAMENTALS **4 Hrs.**

Definition and its historical background, Comparison with other programming languages. Python IDEs **Tokens:** Identifiers, Keyword **Data Types:** Int, Float, Bool, Complex, Str, List, Tuple, Dictionary and Set, Type conversion, Concept of mutability, immutability and reusability **Operators:** Arithmetic, Relational, Logical, Identity and Membership, Input and Output methods Control Flow Conditional statements: If, elif and else, nesting, Iterative or loop statement: for, while, for else and while else, Transfer statements: break, continue, pass.

UNIT 2 FUNCTIONS **3 Hrs.**

Function Declaration and calling, Types of Function arguments, scope of variable, Recursive function Recursive function **Types of function:** Anonymous or lambda, Map, Filter, Reduce, Function aliasing, Nested function, Decorator function, Decorator chaining Generator function

UNIT 3 OBJECT ORIENTED PROGRAMMING **4 Hrs.**

Class, Object, Object reference, Constructor and self-variable, Types of variables, Types of method, Destructor, Composition, Aggregation, Inheritance, Polymorphism, Abstract classes.

UNIT 4 FILE AND EXCEPTION HANDLING **3 Hrs.**

File handling Types of files, writing and reading with text file, important functions of file handling, working with directories, CSV file handling, pickling and unpickling **Exception Handling** exceptions handling using try-except blocks. Raising exceptions and custom exception classes.

Max. 14 Hrs.

COURSE OUTCOMES

CO1- Understand the basic concepts of programming with python.
CO2- Demonstrate proficiency in using different data types, operators, and control structures in Python programs.
CO3- Demonstrate the usage of both built-in and user-defined functions
CO4- Define and use classes in Python to represent attributes and methods, and demonstrate composition, aggregation, inheritance and polymorphism.
CO5- Implement exception handling mechanisms to deal with errors.
CO6- Demonstrate file operations and exception handling.

TEXT/REFERENCE BOOKS

1. John V Guttag. “Introduction to Computation and Programming Using Python”, Prentice Hall of India.
2. Allen Downey, Jeffrey Elkner and Chris Meyers "How to think like a Computer Scientist, Learning with Python", Green Tea Press.
3. Al Sweigart, “Automate the Boring Stuff with Python”
4. Martin C. Brown, “Python: The Complete Reference, Osborne, McGraw-Hill
5. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press

UG_2_L					Computer Programming-II Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

COURSE OBJECTIVES

- To impart the basic concepts of Python programming language
- To be familiar with data structures available in Python.
- To understand the concept of function & object-oriented programming.
- To use file and exception handling for designing applications

List of Experiments:

1. **Language fundamentals**:
   Understanding interpretation, program execution flow, tokens, variables and keywords and identifiers, constants and variables
2. **Input Output Operations:**
   Basic data types, mutability, immutability, Operators, Input /output statements, format specifiers, escape sequences
3. **Decision making, Looping control structures**
   If, elif and else, nesting, Iterative or loop statement: for, while, for else and while else, Transfer statements: break,
4. **Advanced datatypes**
   String, List, Tuples, Dictionary, Set
5. **Functions:**
   Basics of functions, types of arguments
6. **Types of functions**
   Anonymous function, function aliasing, nested function, decorator function and generator function
7. **Object Oriented programming**
   class, object, types of variable and methods, composition, aggregation, inheritance, types of inheritance
8. **File Handling:**
   writing and reading with text file and csv file, working with directories, pickling and unpickling

COURSE OUTCOMES

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

CO1- Apply the concepts of variables, decision making and looping with python.
CO2- Apply various data structures available in Python to solve computational problems.
CO3- Implement advanced function concepts such as decorator chaining and generator functions.
CO4- Design and implement object-oriented solutions to programming problems using composition, aggregation, inheritance and polymorphism.
CO5- Develop Python scripts, for file handling tasks.
CO6- Create and manage files in python.

TEXT/REFERENCE BOOKS

1. John V Guttag. “Introduction to Computation and Programming Using Python”, Prentice Hall of India.
2. Allen Downey, Jeffrey Elkner and Chris Meyers "How to think like a Computer Scientist, Learning with Python", Green Tea Press.
3. Al Sweigart, “Automate the Boring Stuff with Python”
4. Martin C. Brown, “Python: The Complete Reference, Osborne, McGraw-Hill
5. R. Nageswara Rao, “Core Python Programming”, Dreamtech Press

3rd Semester

Course Code					Discrete Mathematical Structures (For CS, ICT)					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs./Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	--	--	100

COURSE OBJECTIVES

- To understand the concept of sets, relations, functions and logic.
- To understand Combinatorics as an analytical method for problem solving.
- To apply graph theory-based modeling to solve real life problems.
- To explain the basics of algebraic structures.

UNIT I: SETS, RELATIONS, FUNCTIONS AND LOGIC Finite and Infinite sets, Counting Principle, Principle of inclusion and exclusion, Mathematical Induction, Relations, Types of Relation, Equivalence relation, Partial Ordered Relations. Functions, types of functions, Logic and Propositional calculus – Simple and Compound Propositions, Basic logical operators. Truth tables. Tautologies, Contradictions, logical equivalence, conditional and bi-conditional statements. Valid arguments and Fallacy. Hasse diagram and Lattice.	**10 Hrs.**
UNIT II: COMBINATORICS Permutation, Combination, Pigeonhole principle, Extended Pigeonhole principle, Recurrence relations, Linear recurrence relations with constant coefficients, Solutions of recurrence relations: second order homogeneous linear recurrence relation, general homogeneous linear recurrence relation.	**10 Hrs.**
UNIT III: GRAPH THEORY AND ITS APPLICATIONS Graphs and related definitions, Sub graphs, Homomorphism and Isomorphism, Paths and Connectivity. Traversable and Eulerian graphs, and Konigsberg Bridge problem, Hamiltonian graphs, Labeled and weighted graphs. Complete, regular and bipartite graphs, Planar graphs, Graph coloring. Four color problem. Directed graphs, Strongly and weakly connected graphs, Trees, rooted trees, sequential representation of directed graphs, Adjacency matrix, Powers of the adjacency matrix.	**12 Hrs.**
UNIT IV: ALGEBRAIC STRUCTURES Group, Semi group, Monoids, Properties of a Group, Composition table for finite Group, Order of a group, Order of its elements, Cyclic Group, Generator, Lagrange's Theorem. Ring, Properties of Rings, Integral Domain, Field.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

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

CO1 - Remember the basics of counting and combinatorics.
CO2 - Understand the basic concepts of sets, relations, functions, logic and be able to determine their properties.
CO3 - Model and solve the real life problem using recurrence relations.
CO4 - Apply Graph theory in related areas such as shortest path problems and network flow problems.
CO5 - Defend and point out fallacious reasoning and propositions in algebraic structures.
CO6 - Model and solve any given engineering problem involving graphs and trees.

TEXT/REFERENCE BOOKS

1. Seymour Lipschutz and Marc Lipson, "Discrete Mathematics, Schaum's Series", McGraw-Hill Education.
2. Kenneth Rosen, "Discrete Mathematics and Its Applications", McGraw Hill Education.
3. Bernard Kolman, Robert Busyb and Sharon C. Ross, "Discrete Mathematical Structures", Pearson.
4. Thomas Koshy, "Discrete Mathematics with Applications", Academic Press Inc.
5. Ralph P. Gramaldi, "Discrete and Combinatorial Mathematics", Pearson.
6. C.L. Liu, D.P. Mohapatra, "Elements of Discrete Mathematics: A Computer Oriented Approach", McGraw Hill Education.

Course Code					Database Management Systems					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To learn fundamental concepts of Database management system
- To study various Database design models and normalization concepts
- To apply the above concepts to optimal Database design for various applications
- To carry out data retrieval and manipulation using SQL.

UNIT 1: Introduction And Database Models File Structure: Concepts of fields, records and files, Conventional file system, Advantages of Database management system, Applications, Data abstraction, Data independence, Three levels of the architecture, Database Users. Entity-Relationship Model, Relational Data Model, Relational algebra operators and syntax.	**10 Hrs.**
UNIT 2: SQL Basics of SQL, DDL, DML, DCL, Primary key, foreign key, unique, not null, check, IN operator, Functions, set operations, sub queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All, view and its type.	**10 Hrs.**
UNIT 3: Normalization And File Organization Importance of a good schema design, Problems encountered with bad schema designs, dependency theory - functional dependencies, Armstrong's axioms for FD's, Minimal covers, 1NF, 2NF, 3NF and BCNF. File Organization: Sequential, Indexed and Random File Organization, indexing structure for index files, hashing for direct files, multi-key file organization and access methods, Storage strategies: Indices, B and B+ trees, hashing.	**10 Hrs.**
UNIT 4: Transactions, Query Processing, Security ACID properties, Schedules, Serializability, Concurrency control protocols, measures of query cost, selection operation, sorting, join. Security: Discretionary and Mandatory Access Control; Audit Trails; Multi-Level Security; Statistical Databases; Data Encryption. Introduction to NOSQL Databases.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the need of database management systems.
CO2 - Explain entity relationship and relational database models.
CO3 - Apply SQL commands in database systems.
CO4 - Analyze normalization and file organization techniques in database systems.
CO5 - Analyze concurrency control in database systems.
CO6 - Determine security levels in database systems.

TEXT/REFERENCE BOOKS

1. A Silberschatz, H F Korth and S Sudarshan, “Database System Concepts”, McGRAW Hill.
2. C. J. Date, A. Kennan, and S. Swamynathan, “An Introduction to Database Systems”, PersonEducation
3. RamezElmasri and Shamkant B Navathe, “Fundamentals of Database Systems”, Addison Wesley

Course Code					Database Management Systems Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- To learn fundamental concepts of Database management system
- To study various Database design models and normalization concepts
- To apply the above concepts to optimal Database design for various applications
- To carry out data retrieval and manipulation using SQL.

LIST OF EXPERIMENTS

1. **Database models:**
   Entity Relationship model, Relational model.
2. **Normalization Techniques:**
   1NF, 2NF, 3NF, BCNF
3. **DDL Commands in SQL:**
   Database and table creation, alteration, defining constraints, primary key, foreign key, unique, not null, check.
4. **DML Commands in SQL:**
   IN operator, inbuilt SQL functions - aggregate functions, built-in functions numeric, date, string functions, set operations, sub-queries, correlated sub-queries in SQL, group by, having, order by, join operations, exist, any, all, and types of views.
5. **DCL Commands in SQL:**
   Transaction control commands, Commit, Rollback, Save point.
6. **Implementation of Embedded SQL, PL SQL Concepts.**
7. **Implementation of Cursors, Stored Procedures, Stored Function, Triggers.**

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Design Entity Relationship and Relational database models.
CO2 - Apply normalization techniques to database models.
CO3 - Apply SQL commands in database systems.
CO4 - Apply transaction control commands in database systems.
CO5 - Apply embedded and PL SQL commands in database systems.
CO6 - Create database systems for real time problems.

TEXT/REFERENCE BOOKS

1. Ivan Bayross, SQL, PL/SQL: “The Programming Language of Oracle”, BPB Publications.
2. A Silberschatz, H F Korth and S Sudarshan, “Database System Concepts”, McGRAW Hill.
3. C. J. Date, A. Kennan, and S. Swamynathan, “An Introduction to Database Systems”, Person Education.

COURSE CODE					Digital Logic and Design					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	-	-	100

PREREQUISITES: Elements of Electrical & Electronics Engineering

COURSE OBJECTIVES

- To learn digital circuit design principles and basic concepts.
- To learn design of combinational circuits as per the application needs.
- To learn design of sequential circuits as per the application needs.

UNIT 1: Binary arithmetic and codes Introduction: Digital Systems; Data representation and coding; Logic circuits; integrated circuits; Analysis; design and implementation of digital systems. Truth table; Basic logic operation and logic gates. Number Systems and Codes: Positional number system; Binary; octal and hexadecimal number systems; Methods of base conversions; Binary; octal and hexadecimal arithmetic; Representation of signed numbers; Fixed and IEEE floating point numbers; Binary coded decimal codes; Gray codes; Error detection and correction codes - parity check codes and Hamming codes.	**8 Hrs.**
UNIT 2: Boolean Algebra and combinational circuits Boolean Algebra & Simplification of Boolean Algebra: Basic postulates and fundamental theorems of Boolean algebra; Standard representation of logic functions; The Map Method; SOP and POS forms; Simplification of switching functions K-map and Quine-McCluskey tabular methods; Synthesis of combinational logic circuits. Combinational Logic Modules and their applications: Decoders; encoders; multiplexers; demultiplexers and their applications; Parity circuits and comparators; Arithmetic modules- adders; sub tractors and ALU; Design examples.	**13 Hrs.**
UNIT 3: Sequential circuits and K-maps Sequential Logic systems: Definition of state machines; state machine as a sequential controller; Basic sequential circuits- latches and flip-flops: SR-latch; D-latch; D flip-flop; JK flip-flop; T flip-flop; Timing hazards and races; Analysis of state machines using D flip-flops and JK flip-flops; Design of state machines - state table; state assignment; transition/excitation table; excitation maps and equations; logic realization; Design examples. Finite State machine design and applications: Designing state machine using ASM charts; Designing state machine using state diagram; Design approaches for Synchronous and asynchronous machines; Registers and Counters; Application examples.	**14 Hrs.**
UNIT 4: Programmable Logic Devices **Logic Families:** Transistor-Transistor Logic (TTL); MOSFET; CMOS. **Programmable Logic Devices:** PLAs and their applications; Sequential PLDs and their applications; State-machine design with sequential PLDs; Introduction to field programmable gate arrays (FPGAs).	**7 Hrs.**
	42 Hrs.

TUTORIALS

Unit 1:	• Case Studies to design different circuits from basic gates. • simplify the circuits using boolean algebra. • Case studies and examples of binary codes and conversion.	**3 Hrs.**
Unit 2:	• Solve problems using K-map for simplification of boolean expression. • Examples to design and solve Encoder, Decoder, MUX, DEMUX, Adder Subtractor.	**5 Hrs.**
Unit 3:	• Real-world examples related to Flip Flops and design of counters. • Design of state machines - state table; state assignment; transition/excitation table; excitation maps and equations; logic realization	**5 Hrs.**
Unit 4:	• Case studies based on PLA, PLD, FPGA	**1 Hrs.**

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- Understand the basics of Number Systems, Boolean algebra and Gates.
CO2- Design combinational circuits using logic gates.
CO3- Construct combinational circuits using multiplexers, demultiplexers, encoders and decoders.
CO4- Evaluate different types of logic families for digital circuit design.
CO5- Design and analyze sequential circuits.
CO6- Design digital logic circuits

TEXT/REFERENCE BOOKS

1. M Morris Mano, "Digital Logic and Computer Design", Prentice Hall Publication
2. A. Anand Kumar, "Fundamentals of Digital Circuits", PHI Learning Pvt. Ltd.
3. Malvino and Leach, "Principle of digital Electronics", McGraw-Hill Education.
4. R.P. Jain, "Modern Digital Electronics", McGraw-Hill.

Course Code					Data Structures					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To learn fundamental concepts of data and file structures.
- To implement various data structures and algorithms.
- To understand function of linear and non-linear data structures.
- To use suitable data structure in variety of applications.

UNIT 1: Introduction Datatypes: primitive vs non-primitive; Data structures: linear vs non-linear; Various Operations performed on DS; Performance Analysis and Measurement: Time and space analysis of algorithms; Asymptotic Notations, Searching and sorting in arrays	**08 Hrs.**
UNIT 2: Linear Data Structures **Stack**: Definition and concepts, Push and Pop operations; Representation in memory; Applications: Infix to prefix and postfix, evaluate postfix, recursion; **Queue**: Definition and concept, Enqueue and Dequeue operations; Representation in memory; Types: circular & priority queue. **Linked list**: Definition and concept, Insert and delete operations, Representation in memory, implementing stack and queue using LL, Types: Singly, Doubly, Circular.	**12 Hrs.**
UNIT 3: Non-Linear Data Structures **Trees**: Definition and concept; Memory representation; Types of trees: Binary, complete binary, full, perfect; Traversal: pre-order, in-order, post-order; Binary Search Trees; AVL trees; Min-Max Heap tree (with insert and delete operations for all trees). **Graphs**: Definition and concept; Memory representation; Types of graphs: complete, connected, directed/ undirected, weighted/ unweighted, cyclic/ non-cyclic; Traversal: breadth and depth first search.	**14 Hrs.**
UNIT 4: Hashing The symbol table, Hashing Functions: Division, Multiplication, Mid Square and Folding; Collision Resolution Techniques: Separate Chaining, Linear Probing, Quadratic Probing and Double Hashing.	**08 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, students will be able to

CO1 - Recall the differences between primitive and non-primitive datatypes.
CO2 - Understand the concepts of linear and non-linear data structures.
CO3 - Apply search, insert and delete operation on data structures.
CO4 - Analyse the time and space complexity of algorithms.
CO5 - Evaluate trade-offs between linear and non-linear data structures.
CO6 - Develop strategies for implementing hash tables and resolving collisions.

TEXT/REFERENCE BOOKS

1. Tanenbaum, “Data Structures using C & C++”, Prentice-Hall International
2. Jean-Paul Tremblay & Paul G. Sorenson, “An Introduction to Data Structures with Applications”, Tata McGraw Hill
3. Sartaj Sahani, “Fundamentals of Data Structures in C++”, Galgotia.Publishers

Course Code					Data Structures Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

PREREQUISITES: Computer Programming - I

COURSE OBJECTIVES

- To learn fundamental concepts of Data and File Structures.
- To implement various data Structures and Algorithms.
- To understand function of linear and non-linear data structures.
- To use suitable data structure in variety of applications.

LIST OF EXPERIMENTS:

Arrays, Pointers and structures:

- Implementation of searching and sorting algorithms on arrays.
- Implementation of pointers and structures.
- Implementation of various operations using strings.

Linear Data structures:

- Implementation of Stack data structure and its applications.
- Implementation of various types of Queue data structure and their applications.
- Implementation of various types of Linked list data structure and their applications.

Non-linear Data structures:

- Implementation of binary tree and its traversals.
- Implementation of Binary search tree.
- Implementation of balanced trees.
- Implementation of various graph traversals: DFS & BFS.

Hashing:

- Implementation of Hash functions and tables.
- Implementation of collision resolution techniques.

COURSE OUTCOMES

On completion of the course, students will be able to

CO1 - Recall the differences between primitive and non-primitive datatypes.
CO2 - Apply searching and sorting algorithms on linear data structures.
CO3 - Implement search, insert, and delete operations on data structures.
CO4 - Implement hash tables and collision resolution techniques.
CO5 - Evaluate the complexity of algorithms.
CO6 - Design application using linear and non-linear data structures.

TEXT/REFERENCE BOOKS

1. Tanenbaum, “Data Structures using C & C++”, Prentice-Hall International
2. Jean-Paul Tremblay & Paul G. Sorenson, “An Introduction to Data Structures with Applications”, Tata McGraw Hill
3. Sartaj Sahani, “Fundamentals of Data Structures in C++”, Galgotia.Publishers

Course Code					Object Oriented Programming Paradigms					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Appraise the appropriate use of OOP Concepts in program design.
- Get acquainted with advanced OOP concepts including streams and multifile programs, exceptions, standard template library.
- Understand the use of performance optimization techniques.
- Understand various design patterns and best practices of Object-Oriented Software Design.

UNIT 1: Introduction to Object-Oriented Programming Overview of Object-Oriented Programming (OOP) principles, Difference between OOP and procedure oriented programming, Classes,, class members and Objects, Encapsulation, Instantiation, Data Hiding, Access Specifiers and their role in encapsulation, Constructors, Destructors and memory management through destructors, Execution order.	**12 Hrs.**
UNIT 2: Inheritance and Polymorphism Introduction to inheritance and polymorphism, Types of inheritance: single, multiple, multilevel, hierarchical, base class and derived class, Hierarchies and Multiple Inheritance, Function Overloading and Operator Overloading, Function overriding, Virtual Functions and Abstract Classes, Friend Functions, Static Function	**12 Hrs.**
UNIT 3: Advanced Concepts using C++ 'this' pointer, Dynamic Memory allocation, Streams and multifile programs, Templets and Exception handling in C++, Standard Template Library (STL) Overview and Usage, Performance optimization techniques.	**10 Hrs.**
UNIT 4: Design Patterns and Best Practices Design Patterns Overview: Creational, Structural, and Behavioural Patterns, Applying Design Patterns in C++, Object-Oriented Design Principles (SOLID), Code Refactoring Techniques, Unit Testing and Test-Driven Development (TDD)	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand object-oriented programming principles.
CO2 - Identify different programming constructs for a problem.
CO3 - Illustrate the use of object-oriented programming constructs.
CO4 - Compose exception handling framework.
CO5 - Select appropriate performance optimization techniques.
CO6 - Appraise design patterns and associated best practices.

TEXT/REFERENCE BOOKS

1. Object Oriented Programming in C, Robert Lafore
2. "The C++ Programming Language" by Bjarne Stroustrup
3. Y.P. Kanetkar ,"Let us C++", BPB Publication
4. "Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides
5. "Programming: Principles and Practice Using C++" by Bjarne Stroustrup
6. "C++ Primer" by Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo

Course Code					Object Oriented Programming Paradigms Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- Hands-on Object-Oriented Programming (OOP)
- Apply advanced OOP concepts including streams and multifile programs, exceptions, standard template library.
- Apply appropriate performance optimization techniques.
- Apply design patterns and best practices of Object-Oriented Software Design.

LIST OF EXPERIMENTS:

1. **Introduction to Object-Oriented Programming**:
   Implement a C++ program with a class representing a simple real-world entity (e.g., Student, Car, Employee). Define class members (attributes and methods) to represent the characteristics and behavior of the entity.
2. **Apply concept of encapsulations:**
   Utilize access specifiers to demonstrate encapsulation and data hiding.
3. **Demonstrate constructors and destructors:**
   Implement constructors to initialize class members and demonstrate the concept of instantiation by creating multiple objects of the class.
   Implement destructors to manage memory resources (e.g., dynamically allocated memory)
4. **Inheritance and its types:**
   Implementation of program to demonstrate inheritance. Types of inheritance: single, multiple, multilevel, hierarchical, base class and derived class, Hierarchies and Multiple Inheritance.
5. **Polymorphism and its types:**
   Implementation of program to demonstrate polymorphism. Types of polymorphism: compile-time polymorphism, run-time polymorphism.
6. **Implement overloading and overriding concepts:**
   Understand and demonstrate through implementation Function Overloading and Operator Overloading, Function overriding
7. **Abstraction and different types of functions in OOP:**
   Illustrate program executing Abstract Classes, Virtual Functions, Friend Functions, Static Function
8. **Pointer, Streams, Templets and multifile programs:**
   Execute the use of This pointer, Streams and multifile programs, Templets
9. **Exception handling:**
   handling of exceptions using the try and except statement.
10. **STL and optimization techniques:**
    Standard Template Library (STL) Overview and Usage and Introduction to performance optimization techniques.
11. **Unit Testing and Test-Driven Development (TDD)**
    Understanding Unit Testing and Test-Driven Development (TDD) and execute basic demonstration programs.

COURSE OUTCOMES

On completion of the course, student will be able to
CO1 - Apply object-oriented programming principles.
CO2 - Apply appropriate programming constructs for a problem.
CO3 - Illustrate the use of inheritance and polymorphism.
CO4 - Compose exception handling framework.
CO5 - Create a project based on OOP Concept.
CO6 - Demonstrate performance optimization, design patterns and associated best practices through a project.

TEXT/REFERENCE BOOKS

1. Object Oriented Programming in C, Robert Lafore
2. "The C++ Programming Language" by Bjarne Stroustrup
3. Y.P. Kanetkar ,"Let us C++", BPB Publication
4. "Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides
5. "Programming: Principles and Practice Using C++" by Bjarne Stroustrup
6. "C++ Primer" by Stanley B. Lippman, Josée Lajoie, and Barbara E. Moo

4th Semester

COURSE CODE					Open Source Technologies					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To create a strong foundation of the individual components of web development.
- To manage the configuration and optimization of every LAMP stack component.
- To learn database design principles, SQL queries, and database management techniques.
- To develop proficiency in server-side scripting and building dynamic web applications.

UNIT 1: Linux Operating System Introduction to Linux Operating System, Linux File System Hierarchy, Essential commands for navigating the file system, File and directory manipulation commands, Understanding file permissions (read, write, execute) and ownership, Using chmod and chown commands to modify permissions and ownership, Shell scripting basics, Overview of shell scripting and its importance, Writing and executing simple shell scripts.	**10 Hrs.**
UNIT 2: Apache Web Server Introduction to Apache web server, Installing apache on Linux platforms, Apache configuration file structure, Apache log files, Apache-related commands, Starting apache for the first time PHP Installation and Configuration. Current and Future version of PHP/MySQL, Installing / Building PHP/MySQL on different platforms with Apache.	**10 Hrs.**
UNIT 3: MySQL Database Introduction to MySQL Database, MySQL database management, installation, configuration, database design, MySQL datatypes, SQL querying, Data definition language, Data manipulation language, Transaction and stored procedures in MySQL, Administration tasks and integration with programming languages.	**10 Hrs.**
UNIT 4: PHP – Server side language Introduction to PHP, PHP Syntax and Variables, Control Structures, Functions and Arrays, String Manipulation and Regular Expressions, Form Handling and Super Global Variables, File Handling, Error Handling and Exception Handling, Object-Oriented Programming in PHP, Database Connectivity with MySQL, Session Management and Cookies, Web Application Development.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Identify the role of each component (Linux, Apache, MySQL, PHP) in the open-source technologies.

CO2 - Understand the impact of Linux file system security within the LAMP stack environment.

CO3 - Illustrate the process of handling HTTP requests and responses using Apache web server.

CO4 - Interpret SQL queries to retrieve and manipulate data stored in MySQL databases.

CO5 - Apply object-oriented programming principles in PHP to develop scalable and maintainable web applications.

CO6 - Analyze database schemas and query execution plans to optimize database performance and query efficiency in MySQL.

TEXT/REFERENCE BOOKS

1. James Lee, “Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP”, Pearson.
2. Jon Duckett, “PHP & MySQL: Server-side Web Development” Wiley.
3. Robin Nixon, “Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites (Learning PHP, MYSQL, Javascript, CSS & HTML5)”, O'Reilly Media.

COURSE CODE					Introduction to Data Science					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Understand issues relating to acquisition, cleaning and loading of data,
- Be able to perform exploratory data analysis using Python,
- Understand the basics of how data can be presented and visualized,
- Become familiar with some very fundamental ethical issues in data science.

UNIT 1 Introduction to Data and Science behind the data Introduction to Data Science, Toolboxes: Python, fundamental libraries for data Scientists. Integrated development environment (IDE). Data operations: Reading, selecting, filtering, manipulating, sorting, grouping, rearranging, ranking, and plotting.	**10 Hrs.**
UNIT 2 Data Collection, Pre-Processing and EDA Data Collection Strategies, Data Pre-Processing Overview, Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization, Descriptive statistics, data preparation. Exploratory Data Analysis data summarization, data distribution, measuring asymmetry. Sample and estimated mean, variance and standard score. Statistical Inference frequency approach, variability of estimates, hypothesis testing using confidence intervals, Box Plots, Pivot Table, Heat Map, Correlation Statistics.	**12 Hrs.**
UNIT 3 Model Development Supervised Learning: First step, learning curves, training-validation and test. Learning models generalities, Regression techniques: Linear/Polynomial regression, Classifications: Naïve Bayes, Logistic Regression, support vector machines	**12 Hrs.**
UNIT 4 Model Evaluation Generalization Error, Out-of-Sample Evaluation Metrics, Cross Validation, Overfitting, Under Fitting and Model Selection, Prediction by using Ridge Regression, Testing Multiple Parameters by using Grid Search, precision, recall, F1 score.	**8 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the fundamental of data science concepts, including Python toolboxes and basic data operations.

CO2 - Understand the principles of supervised learning including its applications, advantages, and limitations.

CO3 - Apply data pre-processing techniques, including data cleaning, integration, transformation, and reduction, to prepare raw data for analysis.

CO4 - Apply statistical inference methods to draw conclusions and make decisions from data analysis.

CO5 - Construct supervised learning models, including regression and classification techniques

CO6 - Demonstrate the techniques to solve real-world data science problems effectively.

TEXT/REFERENCE BOOKS

1. Joel Grus, Data Science from Scratch, O'Reilly Media, Inc.
2. Cathy O'Neil, Rachel Schutt, Doing Data Science, Straight Talk from The Frontline. O'Reilly Media Inc.
3. Introducing Data Science, Davy Cielen, Arno D. B. Meysman, Mohamed Ali, Manning Publications Co.
4. Storytelling with Data: A Data Visualization Guide for Business Professionals,v Nussbaumer Knaflic, Wiley
5. Time Series Analysis 1st Edition, James D. Hamilton, Princeton University Press.

UG_5_L					Design Thinking Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- To immerse students in contemporary perspectives, ideas, concepts, and solutions pertinent to designing and executing projects utilizing principles of design thinking.
- To cultivate a mindset and discipline of systemic inspiration driven by a curiosity to explore novel sources of ideas and unconventional models, particularly beyond their typical professional environments.
- To formulate a concrete, feasible, viable, and relevant innovation project or challenge.

LIST OF EXPERIMENTS:

1. **Introduction to Design Thinking:**
   - Differentiating Design Thinking, Principles of Design Thinking
   - The Basis for Design Thinking, The Design Thinking Team
2. **Design Thinking Workshop Essentials**
   - Design Thinking Workshops and Meetings
   - Exercises and Case-based Discussions
3. **Listening and Empathizing Techniques**
   - Observation Skills
   - Structured Open-Ended Approaches, Exercises and Case-based Discussions
4. **Ideation Tools and Frameworks**
   - Design Thinking Frameworks
   - Ideation Tools like Brainstorming and Innovation Heuristics
   - Overcoming Cognitive Fixedness
5. **Using Diagrams and Maps in Design Thinking**
   - Empathy Map, Affinity Diagram, Mind Map, Journey Map
   - Combining Ideas into Complex Innovation Concepts
6. **Storytelling and Scenario Planning**
   - Improvisation Techniques, Scenario Planning
   - Development and Evaluation of Scenarios, Frog Design and Prototyping
7. **Assessing Bias and Strengthening Communication**
   - Assessing Developer and User Perspectives for Bias
   - Applying Frameworks to Strengthen Communication and sustaining a Culture of Innovation

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- Identify different stakeholders involved in a design thinking project.
CO2- Describe how empathy plays a crucial role in the design thinking process.
CO3- Implement listening and empathizing techniques to understand user needs effectively.
CO4- Assess biases in developer and user perspectives to strengthen communication and foster innovation.
CO5- Evaluate the efficacy of storytelling in communicating design solutions.
CO6- Utilize user observations to create empathy maps and journey maps.

TEXT/REFERENCE BOOKS

1. Roger Martin, "The Design of Business: Why Design Thinking is the Next Competitive Advantage",Harvard Business Press.
2. Hasso Plattner, Christoph Meinel and Larry Leifer, "Design Thinking: Understand –Improve– Apply", Springer.
3. Idris Mootee, "Design Thinking for Strategic Innovation: What They Can't Teach You at Business or Design School", John Wiley & Sons.

Course Code					Probability and Statistics					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITES: Basic knowledge of linear algebra, calculus, and mathematical reasoning.

COURSE OBJECTIVES

- To introduce the fundamental concepts and theorems of probability theory.
- Calculate and interpret measures of central tendency and dispersion for different distributions.
- Apply elements of statistical processes for the solution of real life problems.
- Interpret confidence intervals and hypothesis tests to make inferences about population parameters.

UNIT 1: Fundamentals Of Probability Sample space, Axiomatic definition and properties, Probability set function, Conditional probability, Bayes' rule and independence of events, Random variables, Distribution function, Probability mass and density functions, Expectation, Variance, moments, Moment generating function, Chebyshev's inequality.	**10 Hrs.**
UNIT 2: Random Variable Distributions Uniform, Bernoulli, Binomial, Geometric, Negative binomial, Hypergeometric, Poisson, Exponential, Gamma, Weibull, Beta, Cauchy, Double exponential, Normal, Joint distributions, Marginal and Conditional distributions, Weak Law of large numbers and Central limit theorems.	**11 Hrs.**
UNIT 3: Statistical Inference Sampling Techniques and Estimation: Population, Sample, Parameters, Random sampling with and without replacement, Point estimation, Maximum likelihood estimation (MLE) and method of moments, Confidence intervals for population parameters, Hypothesis Testing: Null and alternative hypotheses, Type I and Type II errors, Common hypothesis tests: t-tests, chi-squared tests, and ANOVA.	**11 Hrs.**
UNIT 4: Curve Fitting And Regression Analysis Independence of random variables, Covariance, Functions of random variables, Measure of central tendency, Curve Fitting, Correlation, Simple Correlation, Partial Correlation, Linear regression, Multiple regression, Residual analysis and diagnostics.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the basic principles of probability theory, including sample spaces, events, and probability axioms.

CO2 - Explain the concept of statistical inference and its role.

CO3 - Apply statistical techniques to real-world data sets using software tools.

CO4 - Implement linear and logistic regression models for predictive analytics.

CO5 - Assess the reliability of statistical inference methods based on probability theory, such as confidence intervals and hypothesis testing.

CO6 - Create and justify mathematical models to represent complex probabilistic phenomena and analyze their implications for decision-making processes.

TEXT/REFERENCE BOOKS

1. Robert V. Hogg, Joseph W. McKean and Allen T. Craig, “Introduction to Mathematical Statistics”, Pearson Education.
2. Paul G. Hoel, Sidney Port, and Charles Stone, “Introduction to Probability Theory”, Houghton Mifflin.
3. K. Rohatgi and A. K. Md. E. Saleh: An Introduction to Probabilityand Statistics, John Wiley & Sons.
4. Boes, D. C., Graybill, F. A., Mood, A. M., Introduction to the Theory of Statistics. McGraw-Hill International Book Company.
5. Ross, Sheldon M., Introduction to Probability Models, Elsevier Inc. AP.

COURSE CODE					Computer Organization and Architecture					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	--	--	100

PREREQUISITE : Digital Logic Design

COURSE OBJECTIVES

- To introduce fundamental concepts and principles underlying computer architecture.
- To explore the components of a computer system, including the CPU, memory, I/O devices, and bus architecture.
- To study the memory hierarchy, understand their organization, performance characteristics, and management techniques.
- To examine the design principles of computer arithmetic units.
- To explore techniques for optimizing computer system performance.

UNIT 1: Processor Architecture Fundamentals X86 Architecture, CPU Registers, Segment Registers, General Purpose Registers, Bus Interface Unit, Address bus, Data bus and control bus, Control Unit, ALU, Memory unit, Architectural overview and comparison between 32-bit and 64-bit processor, Microprocessor Pin Configuration, Microcontroller, Comparison with microprocessor, AI Processor, comparison of AI Processor with regular Microprocessor, ARM processor Architecture, Instruction Set Architecture, Instruction interpretation and execution, MIPS Instructions format, ARM – Instruction Set Architecture.	**10 Hrs.**
UNIT 2: ALU, Micro-operation and RTL Arithmetic and logic unit, Register configuration of Signed Magnitude Addition-Subtraction Algorithm, Register configuration of 2's complement Addition Subtraction Algorithm, Signed Magnitude Multiplication Algorithm, Booth's multiplication algorithm, Addressing Modes, Immediate, Direct, Indirect, PC-relative, Indexed, Subroutine Call, Micro-operation and their RTL specification.	**9 Hrs.**
UNIT 3: Control Unit and Memory Architecture CPU control unit design: Hardwired and micro-programmed design approaches, design of a simple hypothetical CPU. CPU – Memory interconnections, Organization of memory modules, Associative memory, Cache Memory, Memory Mapping Techniques. Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB, Secondary Storage – solid-state drive (SSD), Comparison of SSD with HDD.	**12 Hrs.**
UNIT 4: Pipelining and Multiprocessing Unit Pipelining, pipeline hazards, Design issues of pipeline architecture, Instruction level parallelism and advanced issues, RISC and CISC paradigm, Design issues of a RISC processor, Multiprocessor system and its characteristics, Advance Research Topics in Microprocessor, Parallel processing concepts, Parallelism algorithm for multiprocessor systems, Introduction to GPU Architecture, Comparison between GPU and CPU Architecture, Interconnection network, Cache in multiprocessor systems and related problems, Cache coherence protocols.	**11 Hrs.**
	42 Hrs.

TUTORIALS

UNIT 1:	• Case Studies to differentiate between 32-bit and 64-bit processors. • Exploring Bus interface unit, control unit, segment registers, and general-purpose registers. • Case studies and examples showcasing real-world applications of microcontrollers.	**3 Hrs.**

UNIT 2:	• Real-world examples and applications of different addressing modes in assembly language programming. • Numerical problem solving on different addressing modes. • Exercises to implement subroutine calls using various addressing modes.	**3 Hrs.**
UNIT 3:	• Comparative analysis of hardwired and micro-programmed design approaches in terms of complexity, flexibility, and performance. • Case studies on CPU-memory interconnections and organization of memory modules. • Problem solving on Memory organization. • Real-world examples and case studies illustrating the role of interrupts in process state transitions.	**4 Hrs.**
UNIT 4:	• Examples on pipelining and its advantages in CPU performance. • Exercises on identifying and mitigating pipeline hazards in sample code segments. • Case studies and examples illustrating the differences between RISC and CISC instruction sets. • Exercises on parallel algorithms on multiprocessor systems.	**4 Hrs.**

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand fundamental computer architectures, identifying their strengths, weaknesses, and trade-offs.

CO2 - Classify basic computer arithmetic units, including adders, multipliers, and dividers, using digital logic circuits.

CO3 - Apply knowledge of instruction set architecture to influence the design and functionalities of CPU.

CO4 - Apply the memory management techniques in improvement of computer performance.

CO5 - Differentiate the performance of computer systems using techniques such as pipelining, caching, and branch prediction.

CO6 - Apply critical thinking and problem-solving skills to manage real-world design challenges in computer architecture.

TEXT/REFERENCE BOOKS

1. J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann.
2. M. M. Mano, Computer System Architecture, Prentice Hall of India, New Delhi.
3. William Stallings, Computer Organization and Architecture, Pearson.
4. C. Hamacher, Z. Vranesic, S. Zaky, Computer Organization, McGraw Hill Education.
5. J. P. Hayes, Computer Architecture and Organization, McGraw Hill Education.

UG_5_T					Software Engineering					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

PREREQUISITE: Object oriented programming concepts

COURSE OBJECTIVES

- To gain a deep understanding of the systematic approach to developing, operating, maintaining, and retiring software throughout its lifecycle.
- To exhibit strong communication and interpersonal skills, adhering to professional and ethical principles when working as both team members and leaders in multi-disciplinary teams.
- To apply software engineering principles for product development.

UNIT 1: Introduction and Essence of Modern Software Engineering Introduction, Characteristics of Software, need for software engineering, Software quality attributes, Software product pipelines, Software Development Life Cycle and Process: Waterfall, Iterative models, Exploratory models, RAD, Agile and Scrum, Requirement Analysis, Functional and non-functional requirements, the software requirements document and SRS standards, Requirements Engineering Process: Inception, Elicitation, Elaboration, Negotiation, Specification. Software engineering essence, Essence language, Essence kernel, Using essence kernel in agile development practices, Agile Principles, Agile process models through essence kernel, Large scale complex development Using kernel.	**12 Hrs.**
UNIT 2: Modelling & Design Design principles, Design Patterns, Architecture Versus Design, Modularity, Software Components and Connectors, Architecture Styles. System modelling and UML diagrams.	**10 Hrs.**
UNIT 3: Testing & Quality Management Software Testing Strategies: verification vs validation, Types of testing and sub-testing adopted for various software: unit, integration, validation, system, and acceptance testing, Software Quality Assurance vs control, The ISO 9000 quality standards, CMMI Framework, Test case generations, Cyclometric complexity.	**10 Hrs.**
UNIT 4: Software Project Management Maintenance & Reengineering, Risk management: Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan. Software versioning and Continuous integration, Project management and Risk analysis, Configuration management, Cost analysis and estimation.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

At the end of the course, students will be able to –

CO1 - Understand Unified Modeling Language to specify, construct, and document the artifacts of software products.

CO2 - Understand procedural, non-procedural and object-oriented design approaches for efficient software product construction.

CO3 - Understand software testing life cycle and deduce manual test cases for software product.

CO4 - Identify software and hardware requirements of a project.

CO5 - Analyze software product risks and its management strategies.

CO6 - Assess software quality with ISO and CMMI standards.

TEXT/REFERENCE BOOKS

1. Roger S Pressman, Software engineering A practitioner's Approach, McGraw Hill
2. Ian Sommerville, Software Engineering, Pearson education.
3. Pankaj Jalote, Software Engineering, A Precise Approach, Wiley India.
4. Rajib Mall, Fundamentals of Software Engineering, PHI
5. Unified Modeling Language Reference Manual, James Rumbaugh, Ivar Jacobson, Grady Booch, Addison-Wesley Professional

Course Code					Software Engineering Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

PREREQUISITE: Object oriented programming concepts

COURSE OBJECTIVES

- To foster problem-solving skills and encourage critical thinking by challenging students with real-world software engineering problems and guiding them through the process of finding innovative solutions.
- To raise awareness about ethical and professional responsibilities in software engineering adhering to industry standards and best practices.
- To emphasize the importance of clear and concise documentation throughout the software development process, including requirements documentation, design documents, code comments, and user manuals.

LIST OF EXPERIMENTS:

2. **Requirements Analysis Using Case Modelling:**
   - Conduct interviews or surveys with potential users to gather software requirements.
   - Analyse gathered requirements to identify functional and non-functional requirements.
   - Actor definitions. Writing a case goal.
   - Create use case diagrams, user stories, and requirement documents based on the collected requirements.
8. **Transfer from Analysis to Design in the Characterization Stage**
   - Identify Method, Operation, Object Interface, Class.
   - Finding objects from Flow of Events.
   - Use UML (Unified Modelling Language) to create class diagrams, sequence diagrams, collaboration diagrams, component, activity, and deployment diagrams.
9. **Coding and Implementation.**
   - To Develop small software modules using programming languages like C, C++, Java, Python, or JavaScript.
   - Apply principles of clean code, such as SOLID to write maintainable and readable code.
   - Practice pair programming or code reviews to improve code quality and learn from peers.
10. **Software Testing**
    - Write unit tests using testing frameworks like JUnit, Pytest, or Jasmine.
    - Perform integration testing by combining individual modules and testing their interactions.
    - Conduct system testing to verify that the software meets the specified requirements.
11. **Documentation and Presentation**
    - Create documentation including user manuals, API documentation, and technical specifications.
    - Prepare and deliver presentations to communicate project progress, findings, and recommendations.
    - Practice writing clear and concise technical reports summarizing project activities and outcomes.
12. **Group projects**
    - Participate in a simulated agile development process using Scrum or Kanban boards.
    - Adapt to changing requirements and prioritize tasks based on lab instructor's feedback during iterative development.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Recall fundamental principles and concepts of software engineering.

CO2 - Interpret UML diagrams and understand their significance in software design.

CO3 - Apply software design principles and design patterns to solve specific software design problems.

CO4 - Implement testing techniques to verify and validate software functionality.

CO5 - Synthesize project technical documentation and reports using UML design tools that effectively communicate project progress and outcomes.

CO6 - Design and collaboratively work on group projects to design software architectures and systems that meet specified requirements.

TEXT/REFERENCE BOOKS

4. The Unified Modelling Language User Guide. Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
5. Object-Oriented Software Engineering: using UML, Patterns, and Java. Bernd Bruegge and Allen H. Dutoit.
6. Design Patterns: Elements of Reusable Object-Oriented Software. Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides.

Course Code					Theory of Computation					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	1	0	4	4	25	50	25	-	-	100

PREREQUISITE: Data Structure, Discrete Mathematics

COURSE OBJECTIVES

- Understand automata as a basic model of computation and abstractions.
- Link between formal languages, grammars, automata, and decision problems.
- Understand the Classification of machines by their power/ limitations to recognize languages.
- Understand algorithms and computability through the lens of Turing machines.

UNIT 1: Finite Automata and Regular Language Regular Languages: Alphabets, Language, Regular Expression, Definitions of Finite State automata, Transition Graphs, Conversion of DFA to Regular Expression, Deterministic & Non-deterministic Finite State Automata, Regular Grammar, Regular languages: Closure properties, product construction, Equivalence of NFA and DFA, NFA with epsilon transitions, Thompson's Construction to Convert Regular Expression to NDFA, Subset Algorithm to convert NDFA to DFA, Minimization of DFA, Finite State Machine with output-Moore machine and Mealy Machine, Conversion of Moore machine to Mealy Machine & Vice-Versa, Limitations of Automata Non-regularity: Pumping Lemma and Myhill–Nerode theorem.	**12 Hrs.**
UNIT 2: Pushdown Automata and Context-free Language Context Free Grammar, Derivation tree and Ambiguity, Removing ambiguity, Parse Tress, Application of Context free Grammars, Closure Properties of CFG, Decidable of CFG, Push Down Automata: definition, the Languages of a PDA, Equivalence of PDAs and CFGs, Equivalence of PDAs accepting by final state and by empty stack, Non-determinism in PDAs, Non-equivalence of NPDAs and DPDAs, Pumping Lemma for Context-Free Languages, Ogden's lemma, Normal Forms for CFG's: Chomsky Normal Form and Greibach Normal form, Context–Sensitive Grammars, Linear Bounded Automata.	**12 Hrs.**
UNIT 3: Turing Machine and Recursive Language Universal Turing Machine, Post Machine, Deterministic Turing machines, Multi-tape TMs and its equivalence to single-tape TMs, Nondeterministic TMs and its equivalence to deterministic TMs, Chomsky Hierarchy, Decidable and recognizable languages, Post correspondence problem, Halting Problem, Turing enumerability, Turing machines Acceptability, Decidabilities, and Reducibility, Unsolvable problems about Turing machines, Rice's Theorem, The Church-Turing Thesis.	**09 Hrs.**
UNIT 4: Computability and Complexity The Extended Church-Turing Thesis, Time Complexity, P and NP complexity class, SAT, Polynomial time Reducibility, The time hierarchy theorem, Definition of NP via non-deterministic Turing machines, NP-complete problems, Cook-Levin theorem, NP-complete problems via reductions, PSPACE complexity class, PSPACE-Completeness, PSPACE = NPSPACE, BPP complexity class and relation with the other classes.	**09 Hrs.**
	42 Hrs.

TUTORIALS

1. Problems DFA construction, and conversion of DFA to Regular Expression.
2. Problems on NFA construction and epsilon transitions.
3. Problems on NFA to DFA Conversion.
4. Problems on Minimization of DFA.
5. Problems on Mealy and Moore machine construction, and conversion.
6. Pumping Lemma and Problems on Grammar construction.
7. Problems on parsing and ambiguity removal.
8. Problems on PDA construction, accepted by final state and by empty stack.
9. Problems on NPDA construction and Pumping Lemma for CFL.
10. Conversion from CNF to GNF.
11. Problems on Deterministic and Non-Deterministic TM Construction.
12. Problems on Multi-tape TM and its equivalence conversion.

13. Problems on TM acceptability, decidabilities, and reducibility.
14. Problems on SAT, Polynomial time reducibility, and time hierarchy theorem.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Define key terms and concepts in automata theory, formal languages, and computability theory.
CO2 - Understand the finite state machines for modelling and solving computing problems.
CO3 - Construct finite automata for specific regular languages and prove the properties of regular languages.
CO4 - Analyse context-free grammar to design pushdown automata.
CO5 - Apply the concepts of Turing machine to understand decidability and undecidability.
CO6 - Analyse the probabilistic Turing machine and its bounded error class.

TEXT/REFERENCE BOOKS

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education.
2. Mishra and Chandrashekaran, Theory of Computer Science – Automata languages and computation, PHI.
3. Michael Sipser, Introduction to the Theory of Computation, Cengage Learning.
4. John C Martin, Introduction to Languages and the Theory of Computation, TMH.
5. Sanjeev Arora and Boaz Barak, Computational Complexity: A Modern Approach; Cambridge.

Course Code					Design and Analysis of Algorithms					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Analyse the asymptotic performance of the algorithms
- Implement time and space efficient optimized algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.

UNIT 1: Introduction Elementary Algorithmic concept: Efficiency of Algorithms, Average & worst-case analysis, Elementary Operation Analysis Techniques. Analyzing control structures, Amortized analysis.	**8 Hrs.**
UNIT 2: Recurrence Solving And Divide & Conquer **Recurrence solving**: Intelligent guesswork and proof by induction, change of variable, recurrence tree, master theorem. **DnC**: Binary Search, Merge sort, multiplying large integers, finding the median, Matrix multiplication, Exponentiation	**12 Hrs.**
UNIT 3: Greedy Algorithms And Dynamic Programming **Greedy**: Coin change, Interval scheduling, interval partitioning, job sequencing with deadline, fractional knapsack, MST using Kruskal algorithm, Dijkstra's algorithm, Huffman coding. **Dynamic**: Coin change, 0/1 Knapsack, Floyd-Warshall algorithm, Matrix chain multiplication, Assembly line scheduling, Travelling Salesman Problem.	**12 Hrs.**
UNIT 4: Backtracking, Branch & Bound, Intractability **Backtracking**: 0/1 Knapsack, n-queens problem. **Branch and Bound**: 0/1 Knapsack, Travelling Salesman Problem. **Intractability**: Brief Overview of NP theory, Some NPC problems: Independent set, vertex cover, clique, Hamiltonian cycle; Polynomial time reduction: Reduction of 3 SAT to clique	**10 Hrs.**
	42 Hrs

COURSE OUTCOMES

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

CO1 - Understand need of complexity analysis of the algorithm.

CO2 - Solve Recurrence relations using Master Theorem, Substitution method, and Recurrence tree.

CO3 - Apply different algorithmic strategies to solve computational problems.

CO4 - Identify the appropriate algorithmic strategy for the given problem definition and showcase proficiency in problem solving.

CO5 - Evaluate Classical problems through Backtracking and Branch & Bound techniques.

CO6 - Analyse the complexity of problem classes P, NP, NP-Complete and NP-Hard.

TEXT/REFERENCE BOOKS

1. Charles E. Leiserson, Thomas H. Cormen, Ronald L. Rivest, Clifford Stein - Introduction to Algorithms, MIT Press.
2. Gilles Brassard & Paul Bratley, Fundamentals of Algorithmic, PHI
3. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekharan, Fundamentals of Computer Algorithms, Galgotia.

Course Code					Design and Analysis of Algorithms Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- Analyse the asymptotic performance of the algorithms
- Implement time and space efficient optimized algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.

LIST OF EXPERIMENTS

Basic Programs and their Analysis:

- Selection sort, insertion sort, merge sort, quick sort.
- Linear and binary search.

Greedy:

- Implement fractional knapsack.
- Implement Kruskal's method for MST using union find data structures.
- Implement Interval scheduling.
- Implement Dijkstra's algorithm for single source shortest path.

Divide and Conquer:

- Implement exponential function.
- Implement the Strassen's matrix multiplication.
- Implement program to multiply two large integers.

Dynamic:

- Implement assembly line scheduling.
- Implement chained matrix multiplication.
- Implement Floyd Warshall algorithm for all pair shortest path problem.
- Implement the 0/1 knapsack problem.

Backtracking and Branch & Bound:

- Implement n-queens problem using backtracking
- Implement TSP using branch and bound.

COURSE OUTCOMES

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

CO1 - Understand need of complexity analysis of the algorithm.

CO2 - Apply greedy strategies to solve standard problems.

CO3 - Apply divide and conquer strategies to solve traditional problems.

CO4 - Apply dynamic programming to solve computational problems.

CO5 - Evaluate classical problems through backtracking and branch & bound techniques.

CO6 - Analyse the complexity of problem classes P, NP, NP-Complete and NP-Hard.

TEXT/REFERENCE BOOKS

1. Charles E. Leiserson, Thomas H. Cormen, Ronald L. Rivest, Clifford Stein - Introduction to Algorithms, MIT Press.
2. Gilles Brassard & Paul Bratley, Fundamentals of Algorithmic, PHI.
3. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekharan, Fundamentals of Computer Algorithms, Galgotia.

5th Semester

UG_1_T					Introduction to Artificial Intelligence					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25			100

COURSE OBJECTIVES

- To identify the problems where AI is required and the different methods available.
- To compare different AI techniques available.
- To define and explain learning algorithms and identify problems in game playing.
- To learn Neural Networks and Expert systems

UNIT 1: Introduction To Ai And Searching AI Problems, Intelligent Agents, Types of Learning, Problem Formulation, Basic Problem-Solving Methods. Search strategies, Uniformed Search Strategies, State-Space Search, BFS, DFS, Greedy Best First Search, A* Search, Hill Climbing, and Adversarial Search: minimax and alpha-beta pruning.	**12 Hrs.**
UNIT 2: Foundations And Applications Of Machine Learning Techniques Machine Learning: Representation (features), Linear Transformations, Decision Tree Classifiers, Introduction to Random Forest, Linear Regression, Problem formulations (classification and regression), PCA and Dimensionality Reduction, Nearest Neighbours and KNN, Applications of Machine Learning.	**10 Hrs.**
UNIT 3: Neural Networks: Structure, Learning, And Applications Characteristics of Neural Networks, Artificial Neural Networks: Terminology, Models of Neuron, Topology, Basic Learning Laws, Back Propagation and Gradient Decent, Regularization, Applications of Deep Learning.	**12 Hrs.**
UNIT 4: Introduction To Reinforcement Learning Markov Decision Processes, Introduction to Reinforcement Learning, Policy Search, Dynamic Programming & TD Methods, Bellman Optimality, Q-Learning, SARSA, Applications of Deep Reinforcement Learning.	**8 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Demonstrate understanding of foundational concepts in artificial intelligence and problem-solving methods.

CO2 - Analyse different informed, uninformed and adversarial search techniques in Artificial Intelligence.

CO3 - Understand basics of machine learning including supervised and unsupervised techniques, such as decision tree classifiers, Nearest Neighbours and Linear Regression.

CO4 - Understand of the basics of Neural Networks, including their architecture, components, and applications in AI.

CO5 - Comprehend basics of Reinforcement Learning, policy search, Dynamic programming and Reinforcement Learning algorithms.

CO6 - Analyse real world applications in Machine Learning, Deep Learning and Deep Reinforcement Learning.

TEXT/REFERENCE BOOKS

1. Russell, S.J. and Norvig, P., Artificial Intelligence: A Modern Approach, Pearson Education.
2. Tom Mitchell, Machine Learning: An Artificial Intelligence Approach
3. Reinforcement Learning: An Introduction Book by Andrew Barto and Richard S. Sutton
4. Kevin Night and Elaine Rich, Nair B., “Artificial Intelligence (SIE)”, McGraw Hill.
5. Dan W. Patterson, “Introduction to AI and ES”, Pearson Education.
6. G.Luger, W.A. Sttubblefield, “Artificial Intelligence”, Addison-Wesley Longman.

COURSE CODE					Computer Networks					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To study the TCP/IP layered architecture and various layer functionalities.
- To understand the different routing strategies in the computer networks.
- To understand the error control and flow control mechanism in TCP/IP network.
- To understand the application layer protocols for various networking applications.

UNIT 1: Network Stack and Data Link Layer Introduction to Computer Network and Internet, Layered Architecture (OSI and TCP/IP), Guided and Unguided Media, Framing, Flow Control , Error Control, Media Access Control (ALOHA, CSMA/CD), Ethernet 802.3, Wireless LAN 802.11, Token Ring 802.5, Reliability Issue, Sliding Window.	**13 Hrs.**
UNIT 2: Network Layer Internetworking and Routing: Best effort Service, Switching, Virtual Circuit, IP Addressing, Routing Issue, Distance Vector and Link State Routing, Open Shortest Path First, Routing Information Protocol, Border Gateway Protocol, IPv6 Addressing, Protocol, Transition from IPv4 to IPv6.	**12 Hrs.**
UNIT 3: Transport Layer End to end delivery issues, Reliable data transfer, Congestion control, Traffic engineering and Quality of Service, TCP and UDP.	**11 Hrs.**
UNIT 4: Application Layer Introduction to Application Layer, Application layer protocols: Domain Name System, File Transfer Protocol, Hyper Text Transfer Protocol, Simple Mail Transfer Protocol, Multipurpose Internet Mail Extension.	**06 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the functionalities of OSI and TCP/IP model.
CO2 - Discuss the MAC layer protocol and LAN technologies.
CO3 - Analyze the Distance Vector and Link State Routing protocol strategies in IP network.
CO4 - Compare the IPv4 and IPv6 Addressing schemes.
CO5 - Determine the congestion control mechanism in TCP/IP network.
CO6 - Examine the working and architecture of client server application.

TEXT/REFERENCE BOOKS

1. Andrew S Tenenbaum, “Computer Networks” Pearson Education.
2. Behrouz A Forouzan, “Data Communication and Networking”, McGraw Hill.
3. William Stalling, “Data and Computer Communication”, Pearson Education
4. James Kurose and Keith Rose, “Computer Networking: A Top Down Approach” Pearson Education
5. Pearson Larry Peterson and Bruce Davie, Computer Networks, A Systems Approach, Morgan Kaufmann

COURSE CODE					Computer Networks Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

PREREQUISITE: Computer Programming Language, Data Structure

COURSE OBJECTIVES

- To design the local area networking topologies
- To analyze the traffic load for network architecture
- To understand the working of interior and exterior routing protocols
- To design client server application using socket programming

LAB EXPERIMENTS

- **Network Devices**
  Hubs, L-2 and L-3 Switches, Router, Repeater, Transmission Media
- **Network Topologies Design**
  Network Topologies – Mesh, Star, Bus, Ring, Hybrid
- **Sliding Window Protocol**
  Stop and Wait, Go Back N, Selective Repeat
- **Framing**
  Bit Stuffing, Byte Stuffing, Escape Sequence
- **Error Control**
  Single Bit Error, Burst Error, Checksum, Cyclic Redundancy Check
- **Virtual Local Area Network**
  Virtual Local Area Network, Traffic Load Estimation
- **Loop Avoidance Techniques**
  Spanning Tree Protocol, Rapid Spanning Tree Protocol
- **Interior Routing Protocol**
  Open Shortest Path First, Routing Information Protocol
- **Exterior Routing Protocol**
  Border Gateway Protocol
- **Socket Programming**
  Transmission Control Protocol Socket, User Datagram Protocol Socket.
- **Application Layer Protocol**
  Domain Name System, Hyper Text Transfer Protocol, Simple Mail Transfer Protocol, File Transfer Protocol

COURSE OUTCOMES

On completion of the course, student will be able to
CO1 - Understand the functionalities of networking devices.
CO2 - Demonstrate the working of LAN topologies and virtual local area network.
CO3 - Construct the loop avoidance mechanism in local area network.
CO4 - Implement the interior and exterior routing protocol strategies.
CO5 - Analyze the traffic load for networking protocols.
CO6 - Design client server application using TCP and UDP port.

TEXT/REFERENCE BOOKS

1. Kenneth L. Calvert, Michael J. Donahoo, “TCP/IP Sockets in Java: Practical Guide for Programmers”, Morgan Kaufmann
2. Elliotte Rusty Harold, “Java Network Programming: Developing Networked Applications”, O'Reilly Media
3. James H. Baxter, “Wireshark Essentials”, PACKT Publishing

Subject Code					Compiler Design					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITE: Concepts of data structures, Theory of computation, Programming skills

COURSE OBJECTIVES

- Familiarize with the front-end as well as back-end stages of compiler design, source file, object file and executable file structures and libraries.
- Understand the differences between constructing lexers/parsers by hand
- Get an experience with generating intermediate representations and learn the importance of designing simpler languages
- Observe how original source program is transformed to an optimized target code using code optimization techniques

UNIT 1: Introduction and Lexical Analysis Overview of compiler, Introduction of phases of a compiler, Cousins of the Compiler- Brief discussion on Preprocessor, Assembler, Linker, Loader, Front-end and Back-end of compiler, Lexical analysis and representation, Input Buffering, Token Recognition and Finite Automata, Error Recovery of lexical analysis.	**10 Hrs.**
UNIT 2: Syntax Analysis Introduction, Parsing, Role of Parsers, Context Free Grammars, Top-Down Parsers (RDP, LL), Bottom-Up Parsers (Operator-Precedence, LR), Error recovery strategies	**13 Hrs.**
UNIT 3: Syntax Directed Translation and Intermediate Code Generation Syntax Directed Definition, S-attributed and L-attributed Definitions, Evaluation order, Syntax directed definition and translation schemes, Different Intermediate Forms, Implementation of Three Address Code, Intermediate code for basic constructs of languages	**10 Hrs.**
UNIT 4: Runtime Environments, Code Generation and Optimization Storage allocation strategies, Parameter passing, Type checking, Data flow analysis- Basic blocks and flow graphs, liveliness analysis, local and global optimization techniques- semantic preserving transformations and loop optimizations, Target code generation	**09 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Discuss all phases of compiler and roles of lexical analysis.
CO2 - Understand top-down and bottom-up parsing algorithms and error recovery strategies.
CO3 - Illustrate syntax directed translations and evaluation order in parse trees.
CO4 - Analyse the forms of intermediate representations of a source program during compilation.
CO5 - Identify storage allocation and data flow between procedures.
CO6 - Recognize program transformation techniques and generation of target code.

TEXT/REFERENCE BOOKS

1. Alfred V Aho, M S. Lam, R Sethi, Jeffrey D. Ullman. Compilers-Principles, Techniques and Tools, Pearson.
2. Steven Muchnick, Advanced Compiler Design and Implementation, Morgan Kaufmann.
3. K. Muneeswaran,Compiler Design, Oxford university press.
4. Keith D. Cooper and Linda Torczon. "Engineering a Compiler", Morgan Kaufmann.

COURSE CODE					Compiler Design Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
-	-	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- Familiarize with the front-end as well as back-end stages of language processors
- Understand the differences between constructing lexers/parsers by hand versus using automated generators
- Relate the applications of data structures and theory of computation in compiler design
- Hands-on experience with generating intermediate representations for simpler languages
- Meet the requirements of the realistic constraints of compilers while generating target code

LIST OF EXPERIMENT

1. **Lexical analysis using higher level programming:**
   Program to identify keywords, identifiers and others from the given input file using higher level programming.

2. **Lexical Analysis Tool:**
   Programs based on lexical analysis tool (flex or similar).

3. **Top-down parser:**
   Write a program to implement Recursive Decent Parser (RDP) or similar for given grammar.
4. **Bottom-up parser:**
   Program to implement any one shift-reduce parser.
5. **YACC tool:**
   Programs based on YACC tool and context free grammars.

6. **Intermediate code generation:**
   Program related to intermediate code generation from given expression.
7. **Target code generation:**
   Program to generate target code from the intermediate code.

8. **Demo compiler:**
   Implement a demo compiler.

COURSE OUTCOMES

On completion of the course, student will be able to
CO1 - Identify lexemes and tokens in the given input string using higher level programming language.
CO2 - Demonstrate lexical analysis tool for validation of strings accepted by regular expressions.
CO3 - Experiment top-down and bottom-up parsers for comparative study.
CO4 - Explore parsing tool to derive the strings from context free grammars.
CO5 - Build appropriate intermediate code for programming constructs.
CO6 - Apply code optimization techniques related to target code generation.

TEXT/REFERENCE BOOKS

1. John R. Levine, Tony Mason, Doug Brown, lex & yacc, O'Reilly.
2. AW Appel, J Palsberg, Modern Compiler Implementation in Java, Cambridge University Press.
3. AW Appel, M Ginsburg, Modern Compiler Implementation in C, Cambridge University Press.

COURSE CODE					Operating System					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

PREREQUISITE: Basic C Programming and Data Structures

COURSE OBJECTIVES

- To understand the Operating System (OS) role in the overall computer system and to study the operations performed by OS as a resource manager.
- To understand the scheduling and memory management protocols by OS for processes.
- To understand physical and logical memory system within OS and its management algorithms.
- To understand the concepts of Deadlock, input/output, storage and file management by the operating system.

UNIT 1 Introduction of Operating System and Process Management Operating System Objectives and Functions, Computer-System Organization, Computer-System Architecture, Operating-System Operations, Operating-System Structure, Process Management, Memory Management, Storage Management, Protection and Security, Operating-System Structure, System calls and its Types, Process and Threads, Types of Threads, multicore and multithreading, Process Concept, Operations on Process, CPU Scheduling-Basic concepts and algorithms, Thread scheduling.	**10 Hrs.**
UNIT 2 Synchronization Principles of Concurrency, Interprocess Communication: shared and message passing., Race Condition, Mutual Exclusion, the critical section problem, Peterson's Solution, Fork() and Join(), Synchronization Hardware Support, Mutex locks, Semaphores: Usage and implementation, Classical Problems of Synchronization: Bounded-Buffer Problem, Readers–Writers Problem, Dining-Philosophers Problem, Monitors and its usage in Classical Synchronization Problems.	**10 Hrs.**
UNIT 3 Deadlocks Deadlock Characterization, Deadlock Prevention: Ostrich Algorithm, Deadlock Avoidance: Banker's Algorithm, Deadlock Detection, and Recovery from Deadlock: Resource allocation graph and wait-for-graph.	**10 Hrs.**
UNIT 4 Memory and File Management Memory Management: Address Binding, Swapping, Contiguous Memory Allocation, Segmentation, Paging. Virtual Memory Management: Background, Demand Paging, Copy-on-Write, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing. Storage Management- File and Directory structure, Unix I-node Structure, Disk Structure, Disk Attachment, Disk Scheduling.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to:
CO1 - Recognize the fundamental concepts related to CPU management aimed at enhancing system performance.
CO2 - Understand synchronization techniques to manage concurrent processes effectively within the operating system environment.
CO3 - Understand the memory organization in the CPU and its management by the operating system.
CO4 - Assess turnaround time, waiting time, and response time to conduct a comparative analysis between CPU scheduling algorithms.
CO5 - Apply Banker's algorithm as a preventive measure against deadlock situations within the operating system.
CO6 - Assess disk scheduling strategies to efficiently manage the file system operations within the operating system environment.

TEXT/REFERENCE BOOKS

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts, Wiley.
2. W. Stallings, Operating Systems – Internals and Design Principles, Prentice Hall
3. Sumitabha Das, Unix Concepts and Applications, TMH.
4. Andrew S Tanenbaum, Modern Operating Systems, PHI

COURSE CODE					Operating System Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

PREREQUISITE: Basic C Programming and Data Structures

COURSE OBJECTIVES

- To gain hands-on experience in implementing and understanding system calls in a Unix-like environment.
- To become proficient in navigating the Ubuntu Linux operating system environment.
- To familiarize with the fundamentals of shell scripting using Bash on Ubuntu, including script creation, execution permissions, variables, control structures, and input/output redirection.

LIST OF EXPERIMENTS:

1. **Implementation of Linux basic commands on Unix/Linux**
   - cal, echo, date, ls, lp, man, who, whomi, unname, hostname, bc, uname, uptime, help, cd, mkdir, ls -ltr, bc -l, Unix editor commands: vi, touch, nano, File manipulation commands: cat, grep, rm, touch, cp, mv, cut, head, tail, chmod, wc
2. **Implementation of Shell fundamentals on Unix/Linux**
   - If Else construct, looping concepts, switch case concepts, user defined function and the recursion concepts
3. **Programming simple shell programs using shell fundamentals.**
   - To check the given number is even or odd.
   - To check the given year is leap year or not.
   - To find the factorial of a number.
   - To swap the two integers.
   - To create user defined functions for computing minimum and maximum values from the input by the user using command line arguments.
4. **Implementation of system calls of UNIX operating system in C**
   - fork, exec, getpid, exit, wait, close, stat, opendir, readdir
5. **Simulation of unix commands**
   - cp, ls, grep, rm
6. **Group projects**

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Recall and list basic Linux commands and their functionalities.

CO2 - Explain the working principles of shell fundamentals.

CO3 - Develop and implement simple shell programs to accomplish specific tasks such as checking for even/odd numbers, leap years, factorial calculation, and integer swapping.

CO4 - Implement system calls of UNIX operating system in C programs to create processes, manage files, and retrieve directory information.

CO5 - Implement existing system calls of UNIX system in C language.

CO6 - Design and collaboratively work on group projects to integrate process synchronization, scheduling and deadlock concepts learned throughout the theory sessions and produce functional software artifacts.

TEXT/REFERENCE BOOKS

1. Sumitabha Das, Unix Concepts and Applications, TMH.
2. W. Stallings, Operating Systems – Internals and Design Principles, Prentice Hall
3. Ubuntu Unlished, Matthew Helmke, Addison-Wesley Professional

6^{th} Semester

COURSE CODE					Web Development					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To increase knowledge of available web technologies.
- To decorate the web pages using styling sheet.
- To become familiar with scripting languages.

UNIT 1: HTML & Forms Introduction To HTML, WWW, W3C, web publishing, Common HTML, Tags Physical & Logical, Some basic tags like <body>,changing background colour of page, text colour etc., Text formatting tags, <p> ,<hr> tags, Ordered & Unordered Lists Tags, Inserting image, Links: text, image links, image mapping, Tables, Frames, Form, Introduction with text box, buttons, radio buttons, Working with List box, checkbox, text area	**9 Hrs.**
UNIT 2: Cascaded Style Sheet Introduction to Style sheet, types of style sheets- Inline, External, Embedded CSS, text formatting properties, CSS Border, margin properties, Positioning, Use of classes in CSS, colour properties, use of & CSS background effects, padding properties, image properties	**10 Hrs.**
UNIT 3: Form Validation using Java Script Intro to script, types, intro of JavaScript, JavaScript identifiers, operators, control & Looping structure, Intro of Array, Array with methods, Math, String, Date Objects with methods, User defined & Predefined functions, DOM objects, Event handling, Validations On Forms	**10 Hrs.**
UNIT 4: Server-Side Scripting using PHP Introduction to PHP and the Web Server Architecture Model, HTML Embedding Tags and Syntax, Simple PHP Script, Decision Making, Flow Control and Loops, Working with Arrays Strings and functions, Outputting Data, Include and Require Statements, Built-in Database Functions, connecting to a MySQL, selecting a Database, Building and Sending the Query to Database Engine, Retrieving, Updating and Inserting Data	**13 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Describe the basic html tags.
CO2 - Classify styling sheets as per their use.
CO3 - Apply CSS for web page design.
CO4 - Validate user input using Java Script.
CO5 - Develop server-side scripts using PHP
CO6 - Integrate Back-end with front-end using database connectivity.

TEXT/REFERENCE BOOKS

1. Thomas A. Powell, "Complete reference HTML", Tata McGraw-Hill Publications
2. Dave Mercer, Allan Kent, "Beginning PHP 5", Dreamtech Publication
3. David Flanagn, "JavaScript The definitive guide", SPD Publication

UG_1_T					Object Oriented Programming & Design					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand basics of Object-oriented programming.
- To Create classes for solving real-time problems.
- To implement code reusability using Inheritance.

UNIT 1: Basics OF Object-Oriented Programming Features of Object-Oriented Programming and Java, Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Methods, Overloading, Math class, Arrays in java.	**10 Hrs.**
UNIT 2: Inheritance Basics of objects and classes in java, Constructors, Visibility modifiers, Inbuilt classes in Java, this reference; Inheritance in java, Overriding, Object class, Polymorphism, Dynamic binding, Abstract class, Interface in java, Package in java.	**10 Hrs.**
UNIT 3: I/O Programming, Exception Introduction to Java IO streams, Character and Binary streams, reading data from and writing data to files, Difference between error and exception, Exception handling in Java	**10 Hrs.**
UNIT 4: Event Handling and GUI Programming Event handling in Java, GUI Components, Flow Layouts, Grid Layout, Border Layout	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Define the features of Object-oriented programming.
CO2 - Differentiate between Classes and Objects.
CO3 - Solve code redundancy issue using Inheritance.
CO4 - Distinguish between different types of Polymorphism.
CO5 - Summarize different exception handling mechanisms.
CO6 - Design a graphical user interface using Swing.

TEXT/REFERENCE BOOKS

1. Brett D. McLaughlin, “Headfirst Object-Oriented Analysis and Design”, O Reilly
2. Matt Weisfeld, “The Object-Oriented Thought Process”, Addison-Wesley Professional
3. Herbert Schild, “The Complete Reference, Java 2”, McGraw Hill
4. Balaguruswamy, “Programming with Java – A Primer”, McGraw Hill

24CPXXXT					Cryptography and Network Security					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To learn fundamental concepts of security.
- To understand the mathematical concepts used to build cryptography application.
- To able to apply network security protocols to thwart the security vulnerability.

UNIT 1: Introduction Introduction to Cryptography, Modular Arithmetic, CIA, Classical Cryptography, Substitution and Transposition Techniques, Block Cipher and Stream Cipher, One time Pad, Pseudo Random Number generation.	**10 Hrs.**
UNIT 2: Modern Cryptography Polynomial Arithmetic, AES, Modes of Block Cipher Operations, Types of adversary models and attacks	**10 Hrs.**
UNIT 3: Public Key Cryptography System Mathematical Foundations for PKC; Euler's Totient Function, Fermat and Euler's theorem, Chinese Reminder Theorem, Euclidean and Extended Euclidean algorithm, RSA, Diffie-Hellman Key Exchange, MITM attack	**11 Hrs.**
UNIT 4: Digital Authentication and Network Security Cryptographic hash functions, Message authentication, Digital signature, Kerberos, X.509 authentication service, Email Security: S/MIME, PGP, IPSec, 802.11i.	**11 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand CIA triod

CO2 - Use symmetric and asymmetric key algorithms for designing a security solution.

CO3 - Measure the security strength of a cryptography mechanism.

CO4 - Analyze Key Management techniques and importance of number Theory.

CO5 - Demonstrate the usage of the Message Authentication Codes and Hash Functions for digital authentication.

CO6 - Examine the issues related to network communication and propose solution for it.

TEXT/REFERENCE BOOKS

1. William Stallings, "Cryptography and Network Security Principles and Practice", Pearson Education.
2. Wade Trappe, Lawrence Washington, "Introduction to Cryptography with Coding Theory", Pearson Education
3. Bruce Schneier, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", Wiley Computer Publishing.
4. Wenbo Mao, "Modern Cryptography: Theory and Practice", Prentice Hall.
5. Douglas Stinson, "Cryptography: Theory and Practice", Taylor & Francis.

`

24CPXXXP					Cryptography and Network Security Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	-	-	-	50	50	100

COURSE OBJECTIVES

- To implement modular cryptography of security
- To understand the cryptographic building blocks.
- To apply the network security protocols to thwart the security attacks.

LIST OF EXPERIMENTS:

1. Download and Practice Cryptool.
2. Study and Implement program for Ceaser Cipher with Encryption, Decryption, Brute Force Attack, and Frequency Analysis functions.
3. Study and Implement a program for Transposition (Columnar) Cipher to encrypt and decrypt the message.
4. Study and Implement a program for Rail Fence Transposition Cipher to encrypt and decrypt the message.
5. Study and Implement a program for Vigenère Cipher to encrypt and decrypt the message.
6. Study and Implement a program for 6x6 Playfair Cipher.
7. Study and Implement a program for n-gram Hill Cipher.
8. Use Crypto++ library to implement encryption and decryption functions for different block ciphers.
9. Study and Implement RSA Encryption and Decryption function.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the mapping of mathematical concepts with cryptography algorithms.

CO2 - Learn the modular cryptography building blocksDifferentiate the Symmetric and Asymmetric encryption techniques

CO3 - Evaluate the security strength of a cryptographic algorithm

CO4 - Compare the computational complexity of symmetric and asymmetric cryptography

CO5 - Implement Hashing techniques to create the message digest

CO6 - Utilize the network security tools to analyse network traffic

TEXT/REFERENCE BOOKS

1. William Stallings, "Cryptography and Network Security Principles and Practice", Pearson Education.
2. Wade Trappe, Lawrence Washington, “Introduction to Cryptography with Coding Theory”, Pearson Education
3. Bruce Schneier, “Applied Cryptography: Protocols, Algorithms, and Source Code in C”, Wiley Computer Publishing.
4. Wenbo Mao, “Modern Cryptography: Theory and Practice”, Prentice Hall.
5. Douglas Stinson, “Cryptography: Theory and Practice”, Taylor & Francis.

UG_1_T					Web and Mobile Development Essentials					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
1	-	-	1	1	25	50	25	-	-	100

COURSE OBJECTIVES

- To design web and mobile applications.
- To develop web and mobile applications.
- To introduce Client-side scripting with JavaScript.

UNIT 1: Application Design Introduction to web technology, internet and www, Web site planning and design issues. HTML5: structure of html document, HTML elements: headings, paragraphs, line break, styles, colors, fonts, links, frames, lists, tables, images and forms.	**3 Hrs.**
UNIT 2: Data Validation JavaScript: Overview of JavaScript, Data types, Control Structures, Arrays, Functions and Scopes, HTML5 forms Validation, Objects in JS, DOM: DOM levels, DOM Objects and their properties and methods, Manipulating DOM.	**3 Hrs.**
UNIT 3: Server Programming and Database Connectivity Introduction to server side programming using PHP: Variable, Functions, Object oriented, Loops, Benefits of Database, Types of Database, steps to connect with Database	**4 Hrs.**
UNIT 4: Mobile App Development Android User Interface Design Essentials: User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation, Testing Android applications, Publishing Android application, Using Android preferences, Managing Application resources in a hierarchy, working with different types of resource	**4 Hrs.**
	14 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Describe fundamental concepts of web technologies.
CO2 - Understand the validation mechanism and event handling using JavaScript.
CO3 - Analyse types of databases.
CO4 - Develop mobile application using Android.
CO5 - Use PHP and MySQL to integrate front end and backend.
CO6 - Design Android user interface by efficiently managing the resources.

TEXT/REFERENCE BOOKS

1. POWELL and THOMAS, “Html & Css: The Complete Reference”, McGraw Hill Publisher
2. Laurence Lars Svekis, Maaike van Putten, “JavaScript from Beginner to Professional: Learn JavaScript quickly by building fun, interactive, and dynamic web apps, games, and pages”, Packt Publishing
3. Greg Lim, “Beginning Android Development with Kotlin”, Greg Lim Publisher
4. Steven Holzner, “PHP: THE COMPLETE REFERENCE”, McGraw Hill Education
5. Wallace Jackson, “Learn Android App Development”, APress

PG_1_L					Web and Mobile Development Essentials Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	4	2	4	--	--	--	50	50	100

COURSE OBJECTIVES

- To design web and mobile applications.
- To develop web and mobile applications.
- To introduce client side scripting with Javascript.

LIST OF EXPERIMENTS:

1 HTML page development
Create web page using HTML

2 Designing of Form
Create Form, Document using HTML

3 Form Validation in JS
Validation using java script

4 Implementation of AJAX
Development of AJAX script

5 Server-Side Scripting
Server-side programming Language using PHP, Object Oriented understanding using PHP

6 Database Connection
Implementation of database connection through php

7 Application Development
Mobile Application Development (Android Platform)-I
Mobile Application Development (IoS Platform)-I I

COURSE OUTCOMES

On completion of the course, student will be able to
CO1 - Describe the basic HTML tags.
CO2 - Validate the user input using JavaScript.
CO3 - Construct database for application development.
CO4 - Develop server-side response using PHP.
CO5 - Integrate frontend and backend using database connectivity.
CO6 - Develop Mobile Applications for different platforms.

TEXT/REFERENCE BOOKS

1. POWELL and THOMAS, “Html & Css: The Complete Reference”, McGraw Hill Publisher
2. By Laurence Lars Svekis, Maaike van Putten, “JavaScript from Beginner to Professional: Learn JavaScript quickly by building fun, interactive, and dynamic web apps, games, and pages”, Packt Publishing
3. Greg Lim, “Beginning Android Development with Kotlin”, Greg Lim Publisher
4. Steven Holzner, “PHP: THE COMPLETE REFERENCE”, McGraw Hill Education
5. Wallace Jackson, Learn Android App Development, APress
6. Jesse Feiler, Dummies , iOS App Development For Dummies, Willey

UG_6_T					Distributed Computing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25			100

PREREQUISITE: Operating Systems, Computer Network, Database Systems, Programming Skills

COURSE OBJECTIVES

- Students should gain an understanding of why distributed systems are important, their advantages, and challenges.
- To introduce principles, architectures and algorithms used in distributed systems
- To introduce how large-scale, distributed computational systems are built.
- The focus primarily is to examine state-of-the-art distributed systems.

UNIT 1 Introduction to Distributed Systems & Clocks Definition, Goals, Types, Architectures, Communication- Message-passing systems versus shared memory systems –Primitives for distributed communication –Synchronous versus asynchronous executions –Design issues and challenges, Distributed Computation. **Logical Clocks-**Scalar Time, Vector Time, Physical Clock Synchronization	**10 Hrs.**
UNIT 2 State Machines & Message Passing Global State and Snapshot Recording Algorithms, **Detecting Global Predicate**- Modalities on predicates, Centralized algorithm for relational predicates, Conjunctive predicates, Distributed algorithms for conjunctive predicates**; Election Algorithm**- Leader election in rings, Asynchronous leader election with identities, Synchronous leader election by abusing the synchronous model, **Message Ordering and Group Communication**- Termination Detection Algorithms and Reasoning with Knowledge	**10 Hrs.**
UNIT 3 Consistency, Consensus & Fault Tolerance **Distributed Mutual Exclusion Algorithms**- Lamport's algorithm, Ricart–Agrawala algorithm, and Deadlock Detection Algorithms, Distributed Deadlock Detection Algorithm, **Distributed Shared Memory**- Memory Consistency Models, Shared Memory Mutual Exclusion, Wait-Freedom Algorithm; **Byzantine Fault Tolerance, Checkpointing and Rollback Recovery**- Checkpoint-based recovery and Log-based rollback recovery; **Consensus and Agreement**- : Agreement in a failure-free system (synchronous or asynchronous) , Agreement in (message-passing) synchronous systems with Failures, Agreement in asynchronous message-passing systems with failures , Wait-free shared memory consensus in asynchronous systems;	**13 Hrs.**
UNIT 4 Distributed Storage & Case Studies **Distributed file servers**- GFS, MapReduce; **Selected case studies**- PNUTS, PAXOS, SPARK, Cryptocurrencies; **Advanced Topics**: ML in Distributed Systems-Tensorflow, Parameter Server	**9 Hrs.**
	Max. 42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- Define the core concepts of distributed computing, including its goals, types, and architectural models.

CO2: Understand logical clock models such as scalar time, vector time and clock synchronization algorithms.

CO3: Design message passing algorithms for global state recording, snapshot recording, and detecting global predicates.

CO4: Evaluate distributed deadlock detection algorithms and their effectiveness in preventing system failures.

CO5: Analyse fault-tolerant distributed systems using Byzantine fault tolerance techniques, integrating checkpointing and rollback recovery mechanisms to safeguard system reliability and data integrity.

CO6: Evaluate case studies including PNUTS, PAXOS, SPARK, and cryptocurrencies, and advanced topics such as machine learning integration in distributed systems using frameworks like TensorFlow and Parameter Server.

TEXT/REFERENCE BOOKS

1. Ajay D Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press.
2. Andrew S Tanenbaum, and Van Steen, Distributed Systems: Principles and Paradigms, Pearson Education India.
3. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, Chapman and Hall/CRC.

Additional Reference Material:

1. Abadi, et al (2016). TensorFlow: a system for large-scale machine learning. In Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation (OSDI'16). USENIX Association, USA, 265–283..
2. Mu Li, et al. 2014. Scaling distributed machine learning with the parameter server. In Proceedings of the 11th USENIX conference on Operating Systems Design and Implementation (OSDI'14). USENIX Association, USA, 583–598.
3. Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. 2003. The Google file system. SIGOPS Oper. Syst. Rev. 37, 5 (December 2003), 29–43. https://doi.org/10.1145/1165389.945450
4. Gribble, Steven D. et al. "Scalable, distributed data structures for internet service construction." USENIX Symposium on Operating Systems Design and Implementation (2000).
5. Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System.
6. Robbert van Renesse and Deniz Altinbuken. 2015. Paxos made moderately complex. ACM Comput. Surv. 47, 3, Article 42 et al.. 2008. PNUTS: Yahoo!'s hosted data serving platform. Proc. VLDB Endow. 1, 2 (August 2008), 1277–1288. https://doi.org/10.14778/1454159.1454167

UG_6_T					Distributed Computing Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

PRE-REQUISITE:

- Proficiency in programming languages such as Java, Python, or C++.
- Familiarity with basic networking concepts.
- Proficiency in UNIX/Linux

COURSE OBJECTIVES

- Students should gain an understanding of how distributed systems are designed and developed.
- Provide students with practical exposure to various distributed system technologies, frameworks, and tools commonly used in industry settings.

TOOLS & TECHNOLOGIES

- Message Passing: MPI
- Distributed Storage and Processing: Hadoop MapReduce, HDFS
- Distributed Databases: Apache Cassandra, MongoDB
- Container Orchestration: Kubernetes
- Stream Processing: Apache Kafka, Apache Flink

1. Communication & Message Passing
- Implementing message passing using MPI (Message Passing Interface).
- Designing and implementing simple distributed algorithms for message passing.
- Assignment: Implementing parallel algorithms using MPI.

2. Implementing Byzantine fault tolerance algorithms.
- Designing and implementing consensus algorithms (e.g., Paxos, Raft).
- Assignment: Implementing fault-tolerant distributed applications.

3. Distributed Storage and Processing
- Implementing MapReduce algorithms for distributed data processing.
- Designing and implementing a distributed database application.
- Design and deploy a distributed application using container orchestration.

4. GROUP PROJECT

Max. 42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the core concepts of distributed computing, major tools, technologies and frameworks.
CO2 - Apply Message Passing algorithm for synchronous and asynchronous communication.
CO3 - Apply distributed computing principles to design, develop, and implement distributed applications.
CO4 - Design and deploy distributed applications using virtualization or containerization tools..
CO5 - Evaluate the scalability, reliability, and efficiency of distributed applications under various workload conditions.
CO6 - Collaborate effectively within a team to design, implement, and test distributed system solutions.

TEXT/REFERENCE BOOKS

1. Nancy A. Lynch. Distributed Algorithms. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
2. Andrew S Tanenbaum, and Van Steen, Distributed Systems: Principles and Paradigms, Pearson Education India.
3. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, Chapman and Hall/CRC.

7th Semester

Course Code					Global Data Security and Privacy Laws					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the need of Global Data Security and Privacy Laws.
- To be aware about global data protection regulations and compliance framework.
- To be able to implement data security measures.
- To be aware about the impact of emerging technologies on data privacy and security.

UNIT 1: Introduction to Global Data Security and Privacy Laws Overview of data security and privacy concepts, Introduction to major global data protection regulations (e.g., GDPR, CCPA, LGPD), Understanding the importance of data protection in a global context, Case studies highlighting the impact of data breaches on businesses and individuals	**10 Hrs.**
UNIT 2: Compliance Frameworks and Regulations Key global data protection regulations and frameworks, Comparative analysis of GDPR, CCPA, LGPD, and other relevant laws, Understanding the requirements for data collection, processing, and storage, Practical guidance on achieving compliance with international data security standards.	**11 Hrs.**
UNIT 3: Implementing Data Security Measures Strategies for implementing data security measures in compliance with global regulations, Encryption techniques and best practices for data protection, Data breach response and incident management procedures, Role of cybersecurity technologies in safeguarding sensitive data.	**11 Hrs.**
UNIT 4: Emerging Trends and Future Challenges Exploration of emerging trends in global data security and privacy laws, Impact of emerging technologies (e.g., AI, IoT) on data protection, Addressing challenges posed by cross-border data transfers, Future directions in global data security and privacy regulations and their implications for businesses	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the fundamental concepts of data security and privacy.
CO2 - Interpret the requirements and obligations imposed by GDPR, CCPA, and LGPD on organizations.
CO3 - Apply the principles of data protection to real-world scenarios and case studies.
CO4 - Assess the impact of data breaches on organizations and individuals, and formulate appropriate response strategies.
CO5 - Develop data protection policies and procedures tailored to the requirements of global data security laws.
CO6 - Critically evaluate emerging trends in global data security and privacy laws.

TEXT/REFERENCE BOOKS

1. Paul Voigt and Axel von dem Bussche, "Understanding the GDPR: A Practical Guide to Global Privacy and Data Security Law", published by Wolters Kluwer.
2. Lisa J. Sotto (Editor), "Twentieth Annual Data Privacy and Security Law", published by Practicing Law Institute New York.
3. Peter Carey, "Data Protection: A Practical Guide to UK and EU Law", published by Oxford University Press.
4. Jeff Kosseff, "Cybersecurity Law", published by Wiley.

UG_7_T_OE					Green IT					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Students should gain knowledge of the environmental impact of traditional IT infrastructure, including energy consumption, carbon emissions, e-waste generation, and resource depletion.
- Learn about sustainability principles and how they apply to the design, deployment, and management of IT systems and services.
- Understand relevant environmental policies, regulations, and standards related to IT sustainability, both at the national and international levels.
- Develop practical skills for implementing Green IT strategies, including energy auditing, environmental impact assessment, and sustainability reporting.

UNIT 1 INTRODUCTION TO GREEN COMPUTING Introduction, Environmental Concerns and Sustainable Development, Environmental Impacts of IT, Holistic Approach to Greening IT. Green Devices and Hardware with Green Software.	**11 Hrs.**
UNIT 2 GREEN ENTERPRISE Green Enterprises and the Role of IT - Introduction, Organization and Enterprise Greening, Information systems in Greening.	**11 Hrs.**
UNIT 3 MANAGING AND REGULATING GREEN IT Strategizing Green Initiatives, Implementation of Green IT, Communication and Social media. Regulating the Green IT: Laws, Standards and Protocols .	**11 Hrs.**
UNIT 4 GREEN IT CASE STUDIES Awareness to implementations, Research and Development directions. Worldwide Green IT Case Studies	**9 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Define Green IT with its different dimensions and Strategies.
CO2 - Classify Green devices and hardware along with its green software methodologies.
CO3 - Apply the various green enterprise activities, functions and their role with IT.
CO4 - Analyze the concepts of how to manage the green IT with necessary components.
CO5 - Select the various laws, standards and protocols for regulating green IT.
CO6 - Discuss the various key sustainability and green IT trends.

TEXT/REFERENCE BOOKS

1. Toby J. Velete, Anthony T. Velete, Robert Elsenpeter, “Green IT – Reduce Your Information System’s Environmental Impact While Adding to the Bottom Line”, McGraw-Hill
2. John Lamb, “The Greening of IT – How Companies Can Make a Difference for the Environment”, IBM Press
3. San Murugesan, G.R. Gangadharan, “Harnessing Green IT Principles and Practices”, Wiley Publication
4. Richard Maltzman and David Shirley, “Green Project Management”, CRC Press

UG_4_T					Industry 4.0					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
2	0	0	2	2	50	50	25	--	--	100

COURSE OBJECTIVES

- To impart basic ideas in Industry 4.0.
- To provide students with knowledge of designing Industrial 4.0 Systems for various applications.
- Learn the techniques and tools to offer solutions of Industry 4.0 applications.
- Explore the design and analysis of Industry 4.0 systems tailored for different verticals, enabling students to develop innovative solutions for sustainable and efficient operations.

UNIT I: Introduction and Basic concepts of Industry 4.0 Introduction to Industry 4.0, Definition, General framework, Application areas, Introduction to the Evolution of Industrial revolutions, Key features, Need and benefits, Introduction to Industry 4.0 core technologies: Big data, Advanced Robotics, Simulation, Integration, Internet of Things (IoT), Artificial Intelligence (AI), Cybersecurity, Cloud computing, Additive manufacturing and Augmented Reality. Transformation of industrial processes through the integration of modern technologies such as sensing and actuation, communication, and computational processing.	**6 Hrs.**
UNIT 2: Enabling Technologies for Industry 4.0 Introduction to Enabling Technologies for Industry 4.0, Technology Road Map, Introduction to Connectivity (wired and wireless), Introduction to Networking, Wireless Sensor Networks, Embedded Systems for IIoT (Industrial IoT), Smart and Connected Business Perspective.	**7 Hrs.**
UNIT 3: Introduction to Advanced Technologies and Emerging Trends in Industry 4.0 The Internet of Things (IoT), Characteristics, Physical design of IoT, Logical Design of IoT, Challenges for IoT, Sensors and Actuators for IoT Applications, Introduction to Digital Twin, Introduction to Cyber Physical Systems (Robotics), Cloud Computing, Visualization, Basics of Data Analytics, Artificial intelligence, Concept of Drones/Unmanned Aerial Vehicles (UAVs), Cyber security for Industry 4.0.	**8 Hrs.**
UNIT 4: Case Studies and Application for Industry 4.0 Various application of Industrial IoT: Smart City, Smart Grid, Smart Transportation, Smart Manufacturing, Smart Healthcare	**7 Hrs.**
	28 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 – Understand the core elements and basic technologies for Industry 4.0.
CO2 – Understand the traditional business approach by integrating the data and intelligence.
CO3 – Understand the traditional industries with intelligent and automated machines.
CO4 – Utilize data and intelligence for the development of Smart World.
CO5 – Apply the different computational techniques and algorithms for realizing Industry 4.0.
CO6 – Apply the concept, significance and means to achieve sustainable development.

TEXT/REFERENCE BOOKS

1. The Concept Industry 4.0: An Empirical Analysis of Technologies and Applications in Production Logistics by Christoph Jan Bartodziej
2. Ustundag Alp, and Emre Cevikcan, Industry 4.0: Managing the Digital Transformation, Springer
3. Industry 4.0: Entrepreneurship and Structural Change in the New Digital Landscape, By Springer
4. Bali, V., Banerjee, K., Kumar, N., Gour, S., & Chawla, S. K. (Eds.). Industry 4.0, AI, and Data Science: Research Trends and Challenges.

UG_4_L					Industry 4.0 Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/ Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

PREREQUISITES: Python Programming

COURSE OBJECTIVES

- Gain a basic understanding of Industry 4.0 principles and technologies.
- Learn to design and implement simple IoT systems for everyday applications.
- Explore introductory concepts in advanced technologies like cloud computing and AI.
- Discover practical examples of IoT applications in smart cities, transportation, healthcare, and manufacturing for beginners.

LAB EXPERIMENTS

1. **Introduction to programming -** Basic computations using Python programming
2. **Introduction to interfacing -** Introduction to Interfacing of Basic I/O devices
3. **Internet of Thing Study -** Designing the Publish Subscribe system using MQTT and Paho on Mosquito Broker
4. **Understanding of Interface and Controller-** Demo of Interfacing and controlling LED using MQTT protocol and HIVEMQ broker
5. **Communication protocol –** Demo of Interfacing Bluetooth/ GSM module with Arduino platform
6. **Machine Learning in Industry 4.0 –** Implementing a Basic Machine Learning Model on Arduino
7. **Cloud Computing in Industry 4.0 –** Study of Containerising a python application using docker
8. **Learning of UAVs/Drones –** UAVs application programming using python
9. **Development of application for Industry 4.0 -** Demo of line follower autonomous vehicle.
10. **Collaborative Learning –** Case studies/Project based on topics in Unit 4.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Recall key concepts and principles related to IoT, networking, embedded systems, and data analytics.

CO2 - Demonstrate comprehension of the fundamental technologies and methodologies used in Industry 4.0 applications.

CO3 - Apply theoretical knowledge to design and implement IoT systems and experiments in the laboratory setting.

CO4 - Analyze data collected from IoT experiments and draw conclusions based on observed results.

CO5 - Design and develop projects using industry 4.0 technologies to address real-world challenges.

CO6 - Evaluate the effectiveness and efficiency of industry 4.0 solutions developed during the lab sessions, considering factors such as performance, scalability, and security.

TEXT/REFERENCE BOOKS

1. Moshayedi, Ata Jahangir, Amin Kolahdooz, and Liefa Liao. Unity in Embedded System Design and Robotics: A Step-by-step Guide. Chapman and Hall/CRC
2. Kulkarni, Shrirang Ambaji, Varadraj P. Gurupur, and Steven L. Fernandes. Introduction to IoT with Machine Learning and Image Processing Using Raspberry Pi. Chapman and Hall/CRC
3. Practical Python Programming for IoT by Gary Smart
4. Programming Python, 4th Edition by Mark Lutzs

UG_VII_T					Machine Learning					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To equip students with fundamental machine learning concepts.
- To develop skills of using recent machine learning techniques for solving practical problems
- Understand the methods for evaluating model performance.

UNIT 1: Essential Of Machine Learning Linear Algebra: Vectors, Matrices, Linear Transformations, Eigenvalue and Eigenvectors, Calculus: Derivatives, Gradients and Optimization Techniques Like Gradient Descent, Probability and Statistics: Probability Distributions, Expected Values, Variance, Hypothesis Testing and Bayesian Inference, Multivariate Calculus: Partial Derivatives and Gradients of Multivariable Functions.	**12 Hrs.**
UNIT 2: Supervised Machine Learning History of Machine Learning, Regression & Its Type, Gradient Descent & Its Type, Overfitting and Underfitting, Regularization: Lasso, Ridge, Elastic Net, Kernel Ridge Regression, Support Vector Regression, Generalization, Classification Problems, Decision Boundaries, Decision Tree, Nearest Neighbor Methods, Bayes Optimal Decisions, Naive Bayes Classifier, Logistic Regression, Support Vector Machines (SVM), Support Vector Machines and Large-Margin Classifiers, Supervised Model Evaluation Metrics, Artificial Neural Networks, Imbalanced data handling techniques: SMOTE, ADASYN.	**12 Hrs.**
UNIT 3: Unsupervised Machine Learning K-Mean Clustering, Density-Based Clustering: DBSCAN, OPTICS, Hierarchical Clustering with Advanced Linkage Methods, Spectral Clustering for Non-Convex Clusters, Dimensionality Reduction-PCA, Principal Components Analysis (PCA), EM Algorithm, Gaussian Mixture Models, Unsupervised Model Evaluation Metrics.	**08 Hrs.**
UNIT 4: Semi-Supervised Learning And Ensemble Methods Role of Semi-Supervised Learning, Self-training, Co-Training and Multi-View Learning Approaches, Label Propagation and Graph-Based Methods for Semi-Supervised Learning, Rationale for Ensemble Method, Methods for Constructing an Ensemble Classifier, Bagging, Boosting, Random Forests, Empirical Comparison Among Ensemble Methods.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion this course, students will be able to:

CO - 1 Explain the stages of the machine learning pipeline and classify machine learning into supervised, unsupervised, and semi-supervised learning.

CO - 2 Apply feature engineering techniques to preprocess and manipulate data for machine learning tasks.

CO - 3 Evaluate the performance of the different Machine Learning approaches

CO - 4 Identify and mitigate overfitting and underfitting issues in supervised learning models using techniques like regularization.

CO - 5 Understand the principles of unsupervised learning and apply them to solve real-world problems

CO - 6 Justify the use of ensemble methods and explain the process of constructing ensemble classifiers

TEXT/REFERENCE BOOKS

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education.
2. Christopher M. Bishop, "Pattern Recognition and Machine Learning", by Springer.
3. Amanda Casari, Alice Zheng, “Feature Engineering for Machine Learning”, O’Reilly.
4. Andreas Muller, "Introduction to Machine Learning with Python: A Guide for Data Scientists", Shroff/O'Reilly.
5. M. Gopal “Applied Machine Learning”, McGraw-Hill

Course Code					Machine Learning Laboratory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
0	0	2	1	2	--	--	--	50	50	100

PREREQUISITES:

- Familiarity with programming languages such as Python.

COURSE OBJECTIVES

- To familiarize students with popular tools and libraries used in applied machine learning.
- To develop problem-solving skills and critical thinking abilities to approach and tackle different types of machine learning problems effectively.
- How to evaluate the performance of machine learning models using appropriate metrics and validation techniques

LIST OF EXPERIMENTS

Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students. Preferred Programming Language & Platform: Python/R, TensorFlow.

1. Implement regression algorithms like and understand their performance metrics. Implement linear regression using gradient descent. Train the regression models on sample datasets and evaluate their performance using metrics like Mean Squared Error (MSE) and R-squared. Also explore different regularization techniques (e.g., Lasso, Ridge) and analyze their impact on model performance.
2. Implement classifiers such as Naive Bayes, logistic regression, decision trees, and SVMs. Train the classifiers on sample datasets and evaluate their performance using confusion matrices, accuracy, precision, recall, and F1-score. Visualize decision boundaries and analyze the model's behavior.
3. Apply imbalanced data handling techniques (SMOTE, ADASYN) on a dataset with class imbalance.
4. Implement k-means, hierarchical clustering, DBSCAN clustering etc.
5. Apply PCA for dimensionality reduction and visualize the reduced-dimensional data take suitable dataset to demonstrate the concept.
6. Implement Gaussian Mixture Models (GMM) for density estimation and clustering.
7. Implement ensemble methods such as bagging, boosting, and random forests. Train ensemble classifiers on sample datasets and compare their performance against individual classifiers. Conduct an empirical comparison among ensemble methods and analyze their strengths and weaknesses.
8. Apply machine learning techniques to real-world datasets and problems. Choose a real-world dataset relevant to the students' domain or interests. Perform exploratory data analysis (EDA) and feature engineering. Select appropriate machine learning models and train/test them on the dataset. Evaluate model performance and iterate on the model selection and tuning process.
9. Case study on the application of machine learning in various domains like NLP, Computer Vision, Speech and Audio.
10. Course project:Students are required to submit a course project that involves the development of a Machine learning system to provide a solution to a real-world problem using sample or real-time data sets.

COURSE OUTCOMES

On completion of this course, students will be able to:

CO - 1 Utilize machine learning tools and libraries such as scikit-learn and TensorFlow to develop machine learning models

CO - 2 Demonstrates proficiency in importing and exploring sample datasets using scikit-learn, and performs essential data preprocessing tasks

CO - 3 Evaluate a range of machine learning algorithms along with their implementation

CO - 4 Formulate machine learning problems corresponding to different applications.

CO - 5 Apply appropriate machine learning techniques to solve problems of moderate complexity

CO - 6 Develop machine learning-based solutions to real-world problems, optimize the models learned and report on the expected accuracy that can be achieved by applying the models

TEXT/REFERENCE BOOKS

1. Jeff Prosise " Applied Machine Learning and AI for Engineers", O'Reilly Media, Inc.
2. Sebastian Raschka, Vahid Mirjalili, “Python Machine Learning”, Packt publication
3. Manohar Swamynathan, “Mastering Machine Learning with Python in Six Steps: A Practical Implementation Guide to Predictive Data Analytics Using Python”, Apress.

UG_7_T					Cyber Laws and Ethics					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understanding the difference between physical crime prosecution and cybercrime prosecution.
- To understand the legal aspects of cyber activities and be aware about different cyber laws.
- To be aware about ethical and professional conducts for upcoming digital working atmosphere.
- To understand the basics of digital evidence handling and digital forensics.

UNIT 1: Introduction to Cyber Law and Ethics Overview of Cyber Law: Historical development and key principles, Legal Frameworks: International, national, and regional laws governing cyberspace, Ethical Dimensions: Understanding ethical theories and their application in the digital realm, Case Studies: Analysis of landmark cyber law cases and ethical dilemmas	**10 Hrs.**
UNIT 2: Legal Aspects of Cyber Activities Jurisdiction in Cyberspace: Challenges and solutions, Intellectual Property Rights: Copyright, trademark, and patent laws in the digital age, Cybercrimes and Prosecution: Legal responses to hacking, cyberterrorism, and online fraud, Privacy and Data Protection: Legal frameworks for safeguarding personal information online, IT ACT 2000 and its amendments.	**11 Hrs.**
UNIT 3: Ethical Considerations in Cyber Space Ethical Decision Making: Models and frameworks for ethical reasoning, Digital Citizenship: Responsibilities and rights of individuals in online communities, Cybersecurity Ethics: Balancing security measures with individual privacy rights, Emerging Ethical Issues: AI ethics and tools , algorithmic biases, and the ethics of autonomous systems	**11 Hrs.**
UNIT 4: Digital Forensics Principles and methodologies, Digital Evidence Collection: Techniques for preserving, collecting, and analyzing digital evidence, Forensic Tools and Techniques: Overview of software and hardware used in digital investigations, Case Studies in Digital Forensics: Real-world examples of digital forensic investigations and their legal implications.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the basic law terminology and need of international governance in case of cybercrime.

CO2 - Distinguish between normal cyber behaviour and cybercrime.

CO3 - Understand IT-ACT 2000 and its amendments.

CO4 - Applying ethical conducts in professional life and effectively preserving and exploring intellectual property.

CO5 - Analyse digital forensic cases through to explore real world digital forensic tools and digital evidence handling.

CO6 - Analyse use of cyber laws and ethical code of conduct methods with case studies.

TEXT/REFERENCE BOOKS

1. Vakul Sharma, “Information Technology Law and Practice: Law & Emerging Technology Cyber Law”, Universal Law Publishing.
2. David S. Wall, “Cybercrime: The Transformation of Crime in the Information Age”, Wiley Computer.
3. Anirudh Rastogi, “Cyber Law of Information Technology and Internet”, LexisNexis.
4. Eoghan Casey, “Handbook of Digital Forensics and Investigation” , Elsevier
5. ACM Code of Ethics and Professional Conduct
6. ACM/IEEE Software Engineering Code of Ethics and Professional Practice.

Program Elective Course Track 1

Artificial Intelligence/ Machine Learning Track

					Data Warehousing and Data Mining					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITES: Basic knowledge of linear algebra, probability, and statistics.

COURSE OBJECTIVES

- To understand the principles of Data warehousing and Data Mining.
- To be familiar with the Data warehouse architecture and its Implementation.
- To understand the concept of Analytical Processing (OLAP).
- To characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.

UNIT 1: Introduction To Data Warehousing And Data Mining Data Warehouse Definition, Multidimensional Data Models, Data Warehouse Architecture, Data Granularity, Information Flow Mechanism, Two Classes of Data, Data Lifecycle, Data Flow From Warehouse to Operational Systems, Data Mining Functionalities, Classification of Data Mining System, Data Warehouse v/s Data Mining, Data Mining Process, Major Issues in Data Mining, Knowledge Discovery in Databases (KDD), KDD Process, Online Analytical Processing (OLAP)– OLAP and Multidimensional Data Analysis, On-Line Analytical Processing And Mining	**13 Hrs.**
UNIT 2: Data Mining – Preprocessing Descriptive Data Summarization, Data Cleaning, Integration and Transformation, Data Compression and Dimension Reduction, Principal Component Analysis, Binning Methods, Data Reduction Techniques: Singular Value Decomposition (SVD), Discrete Fourier Transform (DFT), Discrete Wavelet Transform (DWT), Data Discretization and Concept Hierarchy Generalization	**08 Hrs.**
UNIT 3: Association Analysis Basic Concepts, Frequent Item Set Generation, Rule Generation, Various Kinds of Association Rules, Market Basket Analysis, Apriori Algorithm, Eclat Algorithm, FPMax Algorithm, FP-Growth Algorithm, Classification by Decision Tree, Bayesian Classification, Rule Based Classification, Classification by Back propagation, Support Vector Machines, Lazy Learners, Prediction Techniques, Effect of Skewed Support Distribution, Multilevel and Multidimensional, Association Rule Mining v/s Correlation Analysis, Constraint-Based Association Mining.	**12 Hrs.**
UNIT 4: Data Mining On Complex Data And Applications Clustering Algorithms, Cluster Analysis, Algorithms for mining of spatial data, social impacts of data mining, recent trends in data mining, Web Content Mining, Web Structure Mining, Web Usage Mining, Spatial Mining, Multimedia Data Mining, Text Mining,	**09 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of this course, students will be able to:

CO1 - Understand the fundamental concepts of data mining and data warehousing.

CO2 - Identify the need of Data Warehouse tools and techniques for designing and developing different types of databases.

CO3 - Preprocess data for mining by handling missing values, noise, and outliers.

CO4 - Describe the use of Online Analytical Processing to analyze and interpret data

CO5 - Apply various data mining techniques, including association rule mining, classification, clustering, and anomaly detection.

CO6 - Evaluate and interpret the results of data mining algorithms.

TEXT/REFERENCE BOOKS

1. Tan, Pang-Ning, Michael Steinbach, and Vipin Kumar, “Introduction to data mining”, Pearson Education.
2. Prabhu, “ Data warehousing - concepts, Techniques, Products and Application s”, Prentice Hall of India.
3. Alex Berson and Stephen J. Smith, “Data Warehousing, Data Mining & OLAP”, Tata McGraw Hill
4. Gupta, Gopal K., “Introduction to data mining with case studies” PHI Learning Pvt. Ltd

UG_E_AI/MLTrack					Soft Computing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the concepts of feed forward &feedback neural networks
- To understand the concept of fuzziness involved in various systems
- To expose the ideas about genetic algorithm
- To provide adequate knowledge about of FLC and NN toolbox

UNIT 1 : Artificial Neural Network **Introduction:** Feed-forward, recurrent and multi-layer architectures, Supervised and unsupervised learning, **Characteristics:** Adaptability, fault tolerance, generalization, limitations of neuro-computing, **Perceptron:** Linear classifiers; Simple perceptron; Perceptron learning, algorithms; ADALINE; MADALINE; Limitation of perceptron dichotomizer.	**12 Hrs.**
UNIT 2: Multi-Layer Perceptron Network Multi-Layer Perceptron, Gradient decent scheme for error, minimization, Generalized delta learning rule, Back-propagation learning for multi-layer networks, Multi-layer perceptron's for multi- dimensional functional mappings, **Associated Memory Networks:** Auto-association; Hetero-association, **Linear associative networks:** Hebbian learning, perfect recall, cross-talk; Bidirectional associative memory, **Brain-State-in-a-Box network**, **Hopfield Networks:** Binary Hopfield network, basic structure, asynchronous updating, convergence, associative memory, Continuous-valued Hopfield network. Advantages and limitations, **Kohonen Networks:** Self-organizing networks, Similarity measures; Kohonen's winner-take-all network, Geometrical interpretation of Kohonen's learning, Functional specificity of human brain, Kohonen's self-organizing feature map algorithm; Conscience algorithm.	**12 Hrs.**
UNIT 3: Adaptive Resonance Theory ART and stability-plasticity dilemma; ART-1 architecture and algorithm, search, comparison and recognition phases, effect of vigilance, **Radial Basis Function Networks:** radial basis vs. linear basis, Gaussian basis functions, K-means learning, LMS algorithm, comparison with Multi-Layer Perceptron networks, **Support Vector Machines (SVM):** Optimal hyperplane for linear separability, quadratic optimization, SVM for pattern recognition, different kernels for hidden-layer, optimal design of SVM.	**09 Hrs.**
UNIT 4: Fuzzy Network Different faces of imprecision - inexactness, Ambiguity, Undecidability, Fuzziness and certainty, Fuzzy sets and crisp sets, Basic Definition, Fuzzy-set-theoretic Operations, Member Function, Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Fuzzy propositions, Methods of decompositions and defuzzification, Fuzzy-neural networks, Neuro-fuzzy systems, Genetic algorithms: selection schemes, operations, hybrid algorithms,	**09 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Ability to comprehend machine learning and soft computing techniques in solving real world applications
CO2 - Describe the differences between neural networks and fuzzy logic systems.
CO3 - Apply soft computing methods to solve real-world problems.
CO4 - Analyse the strengths and limitations of different soft computing approaches.
CO5 - Evaluate the performance of soft computing models using appropriate metrics.
CO6 - Synthesize knowledge from different soft computing techniques to propose hybrid systems.

TEXT/REFERENCE BOOKS

1. S.R.J ang, C.T. Sun and E. Mizutani, "Neuro-Fuzzy and Soft Computing", Prentice Hall of India and Pearson Education.
2. D.E. Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley, New York.
3. S. Rajasekaran and G.A.V. Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", Prentice Hall of India.
4. R. Eberhart, P. Simpson and R. Dobbins, "Computational Intelligence - PC Tools", AP Professional, Boston
5. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", Wiley India, 3rd edition, 2012.
6. S. Roy and U. Chakraborty, "Introduction to soft computing: neuro-fuzzy and genetic algorithms", Pearson India

UG_1_T					Deep learning					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Understand the basic concepts of neural network.
- Understand the fundamental concepts, usage and impact of neural network, deep learning algorithms in various domains.
- Discuss various deep-learning algorithms to solve real life problems.

UNIT 1: Fundamentals of Neural Networks McCulloch Pitts Neuron, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Feed Forward Neural Networks, Back propagation, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, Singular Value Decomposition, Parameters v/s Hyper-parameters, Learning Rule: Hebb's Rule	**10 Hrs.**
UNIT 2: The Math behind Neural Networks Computation in neural network, The matrix magic, Programmatic expression of deep learning's math constructs, Operations with the tensors, Array broadcasting, Scalar product/Inner product of tensors, Morphing shapes of tensors, Gradient calculation.	**10 Hrs.**
UNIT 3: Data Augmentation and Convolution Neural Network Data Augmentation, CNN, CNN Layers, Kernal, feature map, Pooling: Max Pool, Avg pool, Min Pool, Visualization of 2D convolution, Visualization of 3D convolution,	**10 Hrs.**
UNIT 4: Deep Learning Models Recurrent Neural Network, Long Short Term Memory, Gated Recurrent Unit, Boltzman Machine, Generative Adversarial Network	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Learn how to modify state-of-the-art deep learning architectures for a new dataset/task.

CO2 - Know the basic model types used in deep learning, e.g., Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Generative Adversarial Networks (GANs).

CO3 - Understand the basic concepts of neural networks and deep learning methods.

CO4 - Know the suitability of specific deep learning methods to various real world data domains such as the ones arising from text, images, and videos.

CO5 - Re-train and tune hyperparameters of several classes of deep learning methods, in particular CNNs, RNNs, and GANs, on real-world datasets.

CO6 - Extract patterns from complex real world image and text datasets by using deep learning methods.

TEXT/REFERENCE BOOKS

1. Nikhil Buduma, "Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm", O'Reilly.
2. Ian Goodfellow, YoshuaBengio and Aaron Courville, "Deep Learning", MIT Press.

UG_7_T					Agent-based Learning					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

PREREQUISITE: Machine Learning, Probability and Statistics, Linear Algebra, Deep Learning

COURSE OBJECTIVES

- Comprehensive understanding of fundamental concepts in reinforcement learning, including the agent-environment interaction.
- Develop skills to evaluate and analyze the performance of reinforcement learning models using appropriate metrics such as cumulative reward, convergence rate, and learning stability.
- To gain proficiency in designing and implementing reinforcement learning systems, including defining state and action spaces, designing reward functions, selecting appropriate algorithms, and tuning hyperparameters.
- To demonstrate the ability to formulate problems, design solutions, and interpret results in these domains.

UNIT 1 Introduction to Reinforcement Learning Overview of reinforcement learning (RL) concepts, terminology, and applications. History and evolution of RL, from early research to modern deep reinforcement learning. Basic components of RL: agent, environment, state, action, reward, and policy. Types of RL problems: episodic vs. continuing tasks, finite vs. infinite horizon, deterministic vs. stochastic environments. Exploration vs. exploitation trade-off and its importance in RL. Key algorithms: Multi-Armed Bandits, Markov Decision Processes (MDPs), and Dynamic Programming.	**10 Hrs.**
UNIT 2 Value-Based Methods Introduction to value-based methods for RL, focusing on estimating value functions. Temporal Difference (TD) learning and its connection to dynamic programming. Q-learning: the off-policy TD control algorithm. Deep Q-Networks (DQN) and its extensions for handling large state spaces. Double Q-learning and Dueling DQN architectures for improved stability and efficiency. Experience replay and target networks for stabilizing training.	**10 Hrs.**
UNIT 3 Policy-Based Methods Introduction to policy-based methods for RL, focusing on learning a policy directly. Policy Gradient methods: REINFORCE, Actor-Critic architectures. Advantage Actor-Critic (A2C) and Proximal Policy Optimization (PPO) algorithms. Trust Region Policy Optimization (TRPO) and its successor, Trust Region Policy Optimization (TRPO). Importance of exploration strategies in policy-based methods: entropy regularization and exploration bonuses. Applications and extensions of policy-based methods, such as natural gradient methods and distributional RL	**13 Hrs.**
UNIT 4 Model-Based and Advanced Topics Model-based RL: learning a model of the environment and using it for planning and decision making. Model-based RL algorithms: Model Predictive Control (MPC), Model-Based Policy Optimization (MBPO). Model-based versus model-free methods: trade-offs and applications. Exploration in model-based RL: uncertainty estimation and exploration strategies. **Advanced topics in RL:** meta-learning, hierarchical RL, imitation learning, multi-agent RL. Emerging trends and future directions in RL research and applications.	**9 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand mathematical concepts and fundamental concepts in reinforcement learning, including the agent-environment interaction, Markov decision processes (MDPs), reward signals, and exploration-exploitation trade-offs.

CO2 - Apply a variety of reinforcement learning algorithms to solve sequential decision-making problems, including value iteration, policy iteration, Q-learning, deep Q-networks (DQN), policy gradient methods, and actor-critic architectures.

CO3 - Apply policy gradient methods such as REINFORCE and Actor-Critic architectures to learn policies in RL tasks.

CO4 - Analyze the performance of reinforcement learning models using appropriate metrics such as cumulative reward, convergence rate, and learning stability

CO5 - Evaluate the trade-offs between model-based and model-free reinforcement learning methods, considering factors such as sample efficiency, computational complexity, and generalization capabilities.

CO6 - Formulate reinforcement learning techniques to solve real-world problems in domains such as robotics, game playing, finance, healthcare, and autonomous systems.

TEXT/REFERENCE BOOKS

1. R. S. Sutton and A. G. Barto. Reinforcement Learning - An Introduction. MIT Press.
2. Maxim Lapan. Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more. Packt Publishing.
3. Enes Bilgin. Mastering Reinforcement Learning with Python, Packt Publishing.

UG_7_T					Machine Learning for Cyber Security					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To study the technical perspective of cybercrime and cyber forensics.
- Discuss different machine learning techniques
- Understand the role of machine leaning in cyber forensics
- To study machine learning forensics

UNIT 1: Cyber Crime Introduction, Technical Perspective, Abuse & Misuse of Technologies, Cyber-Crime Challenges, Cybercrimes Classification, Tools and Methods in Cybercrime, Key Loggers and Spyware, Virus, Worms, Trojan Horses, Backdoors, DOS and DDOS Attacks, Buffer Overflow, Attack on Wireless Networks, Phishing, Cybercrime Investigation Process, Digital Forensics Investigation.	**10 Hrs.**
UNIT 2: Adversarial Machine Learning Terminology, The Importance of Adversarial ML, Security Vulnerabilities in Machine Learning Algorithms, Types of Adversarial Attacks, Attack Technique: Model Poisoning, Defense Against Poisoning Attacks, Evasion Attack, Defense Against Evasion Attacks, Mitigating Adversarial Effects, Defending against adversarial attacks	**12 Hrs.**
UNIT 3: Introduction To Machine Learning Introduction- Definition, Dataset, Training and Test Dataset, Standardization and Normalization, Machine Learning Applications in Cyber Security and Cyber Forensics, Classification and Clustering, Supervised Learning, Unsupervised Learning, Evaluation Metrics, Decision Tree, Naïve Bayes, Logistic Regression, Support Vector Machine, Random Forest, K Nearest Neighbour Classifier, K-mean Clustering, PCA, Overfitting and Underfitting.	**10 Hrs.**
UNIT 4: Applications Of Machine Learning In Cyber Forensics Machine Learning Forensics, Intrusion Detection, Anomaly Detection, Host Intrusion Detection, Network Intrusion Detection, Web Application Intrusion Detection, Approaches to Overcome Challenges in Forensics, Approaches to Prevent and Detect Cyber Security Attacks, Adversarial Attacks on Different Security Datasets.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- Understand cybercrime and cyber forensics.
CO2- Compare different types of cyber-attacks and how to detect them.
CO3- Compare different Machine Learning approaches.
CO4- Evaluate the performance of Machine Learning Model.
CO5- Discuss the Machine Learning Forensics.
CO6- Analyze different approaches used to detect and prevent cyber security attacks.

TEXT/REFERENCE BOOKS

1. Clarence Chio, David Freeman, “Machine Learning and Security”, O'Reilly.
2. Soma Halder and Sinan Ozdemir, “Machine Learning for Cybersecurity”, Packt
3. Emmanuel Tsukerman, “Machine Learning for Cybersecurity Cookbook”, Packt
4. Sanjay Misra et al.” Confluence of AI, Machine, and Deep Learning in Cyber Forensics”, IGI Global.
5. Clarence Chio, “Machine Learning & Security: Protecting Systems with Data and Algorithms”, O'Reilly.

.

Program Elective Course Track 2

Industry Track

UG_1_T					Object Oriented Modelling and Design					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Understand Object-Oriented Modelling and Design (OOMD) principles and contrast them with traditional modelling approaches.
- Appraise the appropriate use of OOP concepts in program design within the context of OOMD.
- Get acquainted with advanced OOP concepts such as streams and multi-file programs, exceptions, and standard template library, and understand their relevance in OOMD.
- Understand the use of performance optimization techniques in OOMD to enhance system efficiency.
- Gain insight into various design patterns and best practices of Object-Oriented Software Design, and apply them effectively in OOMD projects.

UNIT 1: Introduction, Modelling Concepts and Class Modelling: Overview of Object-Oriented Modelling and Design (OOMD) principles, Contrasting OOMD with traditional modelling approaches, Understanding the core concepts of OOMD, Modelling Fundamentals: Object and Class Concept in modelling, Introduction to associations and relationships, Encapsulation and abstraction in modelling, Access control and visibility modifiers, Constructors and Destructors in modelling, Introduction to Inheritance and its significance in design, Navigation of class models; Advanced Class Modelling, Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages.	**12 Hrs.**
UNIT 2: Use Case Modelling and Detailed Requirements: Comprehensive Overview; Elaborate Object-oriented Requirements Definitions; System Processes: A Detailed Use Case/Scenario Perspective; Identifying Input and Outputs: The System Sequence Diagram; Identifying Object Behaviour: The State Chart Diagram; Integrated Object-oriented Modelling Approach.	**10 Hrs.**
UNIT 3: Process Overview, System Conception, and Domain Analysis: Comprehensive Overview; Development Stages and Life Cycle; System Conception: Formulating a System Concept; Elaborating the Concept; Crafting a Problem Statement. Domain Analysis: Comprehensive Analysis Overview; Domain Class Model: Detailed Domain State Model; Domain Interaction Model; Iterative Analysis Process.	**10 Hrs.**
UNIT 4: Use Case Realization: Design Discipline in Iterative Development; Object-Oriented Design: Bridging Requirements and Implementation; Design Classes and Class Diagrams; Interaction Diagrams: Implementing Use Cases and Defining Methods; Designing with Communication Diagrams; Enhancing the Design Class Diagram; Package Diagrams: Organizing Major Components; Implementation Considerations for Three-Layer Design.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand object-oriented programming principles.

CO2 - Evaluate different object-oriented modeling techniques and assess their suitability for specific software development scenarios.

CO3 - Develop comprehensive object-oriented models for real-world systems using appropriate design patterns.

CO4 - Critically analyze existing object-oriented systems to identify design flaws and propose improvements.

CO5 - Assess the effectiveness of various object-oriented design patterns in promoting modularity, reusability, and maintainability.

CO6 - Integrate object-oriented design principles with other software engineering paradigms to create holistic software architectures that meet both functional and non-functional requirements.

TEXT/REFERENCE BOOKS

1. "Object-Oriented Analysis and Design with Applications" by Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, and Kelli A. Houston
2. "Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides
3. "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development" by Craig Larman
4. "Object-Oriented Modeling and Design with UML" by Michael Blaha and James Rumbaugh

UG_1_T					UI/UX Design					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITE: Web Technology; Software Engineering; Web Development and Mobile Application Development programming Skills

COURSE OBJECTIVES

- To understand User-Centered Design Principles.
- To gain a comprehensive understanding of UI design principles.
- To gain insights into designing for various platforms including web, mobile, and emerging technologies such as wearables and voice interfaces

UNIT 1 INTRODUCTION TO UI/UX DESIGN Understanding the basics of user interface and user experience design, Human-Computer Interaction (HCI) Principles	**10 Hrs.**
UNIT 2 USER RESEARCH AND PROTOTYPING Learning various user research techniques including interviews, surveys, and usability testing, Introduction to prototyping tools and techniques for creating wireframes and mockups.	**11 Hrs.**
UNIT 3 VISUAL DESIGN AND INTERACTION DESIGN Understanding color theory, typography, and layout in interface design, Designing intuitive and interactive user interfaces.	**11 Hrs.**
UNIT 4 ADVANCED TOPICS AND PROJECT MANAGEMENT Designing interfaces for various devices and screen sizes, Developing UX strategies and managing UI/UX projects effectively.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand visual design principles and their application in UI/UX

CO2 - Comprehend HCI principles and their role in designing intuitive interfaces

CO3 - Demonstrate the ability to conduct user research using various methods such as interviews and surveys.

CO4 - Apply principles of typography, color theory, and layout to create visually appealing and cohesive interface designs.

CO5 - Develop skills in wireframing and prototyping

CO6 - Demonstrate proficiency in applying user interface (UI) and user experience (UX) design principles to create intuitive, user-friendly digital experiences

TEXT/REFERENCE BOOKS

1. Jesse James Garrett, The Elements of User Experience: User-Centered Design for the Web and Beyond, New Riders
2. Dan Saffer, Designing for Interaction: Creating Innovative Applications and Devices, New Riders
3. Saul Greenberg, Sketching User Experiences: The Workbook, Morgan Kaufmann
4. Robin Williams, The Non-Designer's Design Book, Peachpit Press Publications
5. Alan Cooper, About Face: The Essentials of Interaction Design, Wiley
6. Ethan Marcotte, Responsive Web Design, A Book Apart
7. Lean UX: Designing Great Products with Agile Teams by Jeff Gothelf and Josh Seiden.

24CSXXXT					Secure System Engineering					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To learn fundamental concepts of secure systems
- To study various system vulnerabilities
- To study various software based vulnerabilities
- To study various application based vulnerabilities

UNIT 1: Introduction Introduction to gdb, buffer overflow, Preventing buffer overflow based malware, Return-to-libc attack, ROP Attacks	**10 Hrs.**
UNIT 2: Software based vulnerabilities Integer overflow and buffer overread and heap overflow, More on heap overflow; Access Control	**10 Hrs.**
UNIT 3: Application based vulnerabilities Confinement, SGX and Trustzone, Software fault isolation, Confinement in Applications, Mandatory access Control	**10 Hrs.**
UNIT 4: Hardware Security Micro-architectural Attacks, Hardware Security, Side Channel Analysis, Protecting against Hardware Trojans, Detecting Hardware Trojans in ICs	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand need of secure systems
CO2 - Explain hardware vulnerabilities models
CO3 - Apply mitigation techniques in systems
CO4 - Analyze security techniques in systems
CO5 - Analyze application vulnerabilities in systems.
CO6 - Determine attack analysis in systems

TEXT/REFERENCE BOOKS

1. "Introduction to Computer Security" by Matt Bishop, Pearson Publications.
2. Security Engineering: A Guide to Building Dependable Distributed Systems by Ross J. Anderson
3. NPTEL course : https://archive.nptel.ac.in/courses/106/106/106106199/

UG_1_T					Agile and DevOps					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Learn about the DevOps lifecycle and its stages, including continuous integration, continuous delivery, and continuous deployment.
- Gain practical knowledge of version control systems and configuration.
- Understand the importance of security in DevOps and learn how to integrate security practices into the CI/CD pipeline.
- Gain insights into sustainable source code management practices, to create longterm value.

UNIT 1: Understanding of Agile Process Software processes, Process Models – overview Agile Model and Various Agile methodologies - Scrum, XP, Lean, and Kanban. Scope of each model and their comparison in real-world case studies.	**10 Hrs.**
UNIT 2: Introduction to DevOps Definition and principles of DevOps, Benefits of DevOps adoption, DevOps culture and mindset, DevOps Lifecycle, Overview of the DevOps lifecycle stages, Continuous Integration (CI) and Continuous Delivery (CD), Automated testing and deployment, Version Control and Configuration Management, Introduction to version control systems (e.g., Git), Branching and merging strategies, Configuration management tools (e.g., Ansible, Puppet)	**10 Hrs.**
UNIT 3: Integration and Build Automation Introduction to Infrastructure as code (IaC) concepts and benefits, Infrastructure provisioning tools (e.g., Terraform, Cloud Formation), Managing infrastructure configurations, setting up CI/CD pipelines, Building and packaging applications, Automated testing and code quality checks, Continuous Deployment and Release Management, Strategies for releasing software updates, Managing deployment environments, Release orchestration and rollback strategies	**10 Hrs.**
UNIT 4: SCM and GIT Fundamentals and Security Overview of source code management and its role in software development, Version control systems and their benefits, Introduction to centralized and distributed SCM, Understanding repositories, commits, and revisions, Exploring the concepts of working directory, staging area, and remote repositories, Introduction to branching and tagging, Introduction to Git and its key features, Setting up a Git repository, Committing changes and viewing history, Working with remote repositories, Cloning, pushing, and pulling changes, Collaboration workflows and managing multiple contributors. Importance of monitoring in DevOps, implementing application and infrastructure monitoring, Log aggregation and analysis, Security and Compliance, DevOps security principles and practices, Implementing security controls in CI/CD pipelines, Compliance considerations in DevOps.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1: Understand the principles and benefits of DevOps, including its role in software development and deployment, enabling them to comprehend its significance in modern software engineering practices.

CO2: Develop an understanding of compliance requirements and learn how to incorporate them effectively into DevOps processes, ensuring adherence to regulatory standards and organizational policies.

CO3: Develop a comprehensive understanding of source code management concepts, including its objectives (such as version control and collaboration) and components (such as repositories, branches, and commits), facilitating efficient and organized software development workflows.

CO4: Aain an understanding of management principles and techniques relevant to source code, including source code planning and optimization strategies, enabling them to effectively oversee and optimize the development process.

CO5: Acquire knowledge of source code management practices and their impact on the team's environment, allowing them to assess, implement, and adapt best practices for efficient collaboration, version control, and code quality assurance within the development team.

CO6: Develop proficiency in utilizing DevOps tools and technologies to automate software development, testing, and deployment processes, enabling students to streamline workflows, increase productivity, and enhance the overall efficiency of software delivery pipelines.

TEXT/REFERENCE BOOKS

1. Gene Kim, Kevin Behr, and George Spafford, "The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win",Revolution Press.
2. Jez Humble and David Farley, “Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation", Addison-Wesley Professional.
3. Emma Jane, “GIT for Teams”, O'Reilly Media.

UG_1_T					Web Testing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Understand applicable theories connecting cybersecurity to human behaviour.
- Critique the purpose and goals of vulnerability assessments and penetration testing.
- Interpret and differentiate cyber threats and exploits in a penetration testing context.
- Describe and distinguish key phases of ethical hacking: reconnaissance, scanning, gaining access, maintaining access, and covering the tracks.

UNIT 1 Principle of Security Testing Explain secure applications and the needs for testing them; model and methodologies to different applications. Different system, recognise project security threats. The following are the tenets of security testing: non-repudiation, availability, authenticity, authorization, and confidentiality. Network security, system software security, client-side application security, and server-side application security are the main areas of attention for security testing.	**10 Hrs.**
UNIT 2 Types of Security Testing Vulnerability Scanning: programme that automatically examines a system to find patterns of known vulnerabilities Security scanning reduces risks and errors by identifying network and system flaws. Security scanning, both automated and human Penetration testing involves simulating a hostile hacker's attack in order to look for any weaknesses that a hacker would use to try to access the system. Risk assessment includes security risk testing, risk classification (low, medium, and high), and risk-reduction strategies. To evaluate and enhance a product's, process', or project's security, gather and evaluate metrics using statistical techniques.	**11 Hrs.**
UNIT 3 Web application testing Network scanning, Determining different attacks on web application, applying the attacks as SQLi, XXS, Session Hijacking, etc and also developing different methodologies to secure application from such attacks and vulnerability present.	**11 Hrs.**
UNIT 4 Introduction to Mobile Security Overview of mobile security challenges and threats, Differences between mobile and traditional computing security, Legal and ethical considerations in mobile penetration testing, Understanding the architecture of mobile applications, Overview of mobile platforms (iOS, Android) and their security models, Introduction to mobile app components (UI, backend, APIs), Understanding static and dynamic analysis of mobile testing	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the fundamental principles of and concepts of mobile and web application penetration testing.

CO2 - Identify different types of attacks of different applications.

CO3 - Apply different principals of security testing.

CO4 - Analyze the potential impact of identified vulnerabilities on the security posture of mobile and web applications.

CO5 - Evaluate the security posture of mobile and web applications using penetration testing methodologies.

CO6 - Create strategies to mitigate security risks and protect against common mobile and web application security threats.

TEXT/REFERENCE BOOKS

1. Gary McGraw, Software Security: Building Security, Addison-Wesley
2. 2. Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw, and Nancy Mead, Software Security Engineering: A Guide for Project Managers, Addison-Wesley
3. Research papers and articles from relevant conferences and journals (e.g., IEEE Transactions on Information Forensics and Security, ACM Transactions on Embedded Computing Systems).
4. Online resources and tutorials on specific hardware security topics.

Program Elective Course Track 3

Speech, Vision & Text Track

					Computer Graphics					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To introduce concept of computer assisted picture generation and manipulation
- To understand an overview of interactive computer graphics, two-dimensional system.
- To understand the most important algorithm for graphical primitives, transformation, clipping and filing for 2D objects
- To study curve generation, 3D picture generation, transformation and animation techniques

UNIT 1: Basic of Computer Graphics and Primitive Drawing Algorithms Basics of Computer Graphics, Applications of computer graphics, Display devices, Graphics input devices, basic primitive drawing algorithms- point, line, circle, ellipse, and polygon drawing algorithms, polygon filling algorithms	**11 Hrs.**
UNIT 2: Two-dimensional Transformation and Viewing Two-dimensional transformations (translation, rotation, scaling, reflection, shearing), matrix representation, composite transformations, homogeneous coordinates, viewing transformation, line clipping, polygon clipping	**11 Hrs.**
UNIT 3: Three-dimensional Object Representations and Transformations Three-dimensional object representations, Three-dimensional transformations (translation, rotation, scaling, reflection, shearing), projection and types of projection, projection transformations, modelling curves, curved lines and surfaces, Visible surfaces	**11 Hrs.**
UNIT 4: Rendering and Colour Models Polygon Rendering methods, Texture mapping, Colour models (RGB, YIQ, CMY), Modelling techniques and fractals, Animation: Computer assisted animation and real time animation techniques	**09 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Discuss the concepts of computer graphics and hardware devices.
CO2 - Compare the algorithms for 2D graphics primitives and attributes.
CO3 - Apply composite geometric transformations and viewing on 2D objects.
CO4 – Represent 3D objects with transformations and projections.
CO5 - Demonstrate the concepts of clipping and visible surface detection in 2D and 3D viewing.
CO6 - Differentiate rendering techniques, colour models and animation techniques for object representation.

TEXT/REFERENCE BOOKS

1. John F. Hughes, Andries van Dam, Morgan McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akeley. Computer Graphics: Principles and Practice, Pearson education
2. David F. Rogers, Mathematical elements for computer graphics, Tata McGraw Hill
3. Donald Hearn, Pauline Baker, Computer graphics with OpenGL, Pearson education

UG_6_T					Digital Image Processing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To introduce mathematical modelling of transforms for digital image processing.
- To understand methodology to analyse, design and implement images processing and analysis algorithms.
- To develop understanding for image preprocessing methodologies like segmentation, noise elimination and edge detection.

UNIT 1: Fundamental And Spatial Domain Processing Introduction, Image sampling and quantization, Basic relationships in pixels, Basic intensity transformations, Histogram processing: local and global, Spatial filtering: smoothing and sharpening, Image Enhancement.	**10 Hrs.**
UNIT 2: Frequency Domain Processing Sampling 2D functions, 2D Fourier Transform and properties, Filtering in frequency domain: smoothing, sharpening. Performance comparison between band pass, gaussian and notch filters in both high and low pass filtering applications.	**10 Hrs.**
UNIT 3: Image Restoration And Morphological Operations Model of image degradation process, Restoration in presence of noise, estimating degradation function, Inverse filtering, MMSE filtering, Morphological operations: Erosion and dilation, opening and closing, Performance comparison between morphological and restoration techniques algorithms for noisy images.	**10 Hrs.**
UNIT 4: Image Segentation And Description Point, line and edge detection, Thresholding, Basic segmentation algorithms: region based, watershed, Image representation: chain codes, polygonal approximation, Boundary and regional descriptors: textures, moments, Feature extraction using shape features. Performance comparison between thresholding and region-based segmentation algorithms.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand properties of an image and its digitization.
CO2 - Understand image processing in spatial and frequency domain.
CO3 - Apply image sharpening and smoothing for image enhancement.
CO4 - Apply spatial and frequency domain filtering for image denoising and restoration.
CO5 - Demonstrate morphological operations like opening and closing on images for connected component extraction in real world applications.
CO6 - Apply edge detection algorithm and thresholding for object segmentation for real world applications.

TEXT/REFERENCE BOOKS

1. Gonzalez, R. C., & Woods, R. E., “Digital image processing”, Pearson.
2. Sonka, Milan, “Image processing, analysis and machine vision”. Cengage Learning Pvt. Ltd.
3. Jayaraman, “Digital Image Processing”. McGrawhill.
4. Gose, Earl, “Pattern recognition and Image Analysis” PHI Learning Pvt. Ltd.
5. Alasdair, McAndrew, “A Computational Introduction to Digital Image Processing”. CRC Press.
6. Artyom M ,Grigoryan, “Image Processing”. Taylor & Francis Ltd.
7. Castleman, Kenneth “Digital Image Processing”. Pearson Education.

24BCPXXXT					Computer Vision					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITE: Linear Algebra

COURSE OBJECTIVES

- Ability to work with features above the pixel level.
- Develop ability to understand the difference in theory and practice of Computer Vision.

UNIT 1: Low Level Vision Introduction, Applications and Challenges, Fundamentals of Image Formation, Pin hole camera, Camera Parameters, Geometric camera calibration, Transformation: Orthogonal, Euclidean, Affine, Projective, Convolution and Filtering, Image Enhancement, Histogram Processing, Stereo Vision, Photometric Stereo.	**10 Hrs.**
UNIT 2: Mid Level Vision Edge and corner detection, Hough Transform, SIFT, SURF, HOG, Image Segmentation: Region Growing, Edge based methods, Graph-Cut, Mean-Shift, K-Means, K-Medoids, EM algorithm.	**10 Hrs.**
UNIT 3: High Level Vision Error, Loss, Image Classification, Object detection: Face Detection, State of the art in Object detection, Object Recognition, Hidden Markov Model, Object Tracking, Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo, Motion parameter estimation.	**10 Hrs.**
UNIT 4: Computer Vision Using Deep Learning Convolutional Neural Networks architectures, Convolution and Pooling, training CNN, Data Augmentation, Transfer Learning, Recurrent Neural Networks, LSTM, GRU, Applications like Image Captioning, Face Recognition, Semantic Segmentation.	**12 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Define low level to high level vision.

CO2 - Describe basic methods of image formation and image processing including camera projections, image filtering, and feature extraction.

CO3 - Develop applications like classification, semantic segmentation, tracking, person identification.

CO4 - Choose appropriate computer vision method for a given problem statement.

CO5 - Apply computer vision techniques for solving practical problems.

CO6 - Implement vision models based on deep neural networks.

TEXT/REFERENCE BOOKS

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited 2011.
2. D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, Pearson Education, 2003.
3. Simon Prince, Computer Vision: Models, Learning and Interface, Cambridge University Press.

COURSE CODE					Natural Language Processing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the structure and basic operations of Natural Language Processing
- To understand the syntax of Natural languages for grouping local words for parsing
- To understand the concepts of linguistic rules and machine learning approaches for classification
- Apply language modelling techniques for improving application of NLP.

UNIT 1: Introduction to Natural Language Processing Building Blocks of NLP- NLP and Ambiguity, POS Tagging, Named Entity Recognition, Word Sense Disambiguation, Wordnet and Lexical Resources, Syntax – Constituency Parsing, Dependency Parsing, Distributional Semantics, Lexical Semantics	**10 Hrs.**
UNIT 2: Classical Text Processing Basic Text Processing, Language Modelling, Advanced smoothing for language modelling, Minimum Edit Distance, Spelling Correction, Models for Sequential Tagging – MaxEnt, CRF, NaiveBayes	**12 Hrs.**
UNIT 3: Deep Learning for NLP & LLM Representation Discovery: Word vectors, Word2Vec & GloVe, RNNs for Variable Length Sequences, Attention & Transformer, Neural & Pre-Trained Language Models, Advanced Pre-training for Language Models, GPT3 & Beyond: Few-Shot Learning, Prompt Learning	**10 Hrs.**
UNIT 4: NLP Applications Sentiment Analysis and Opinion Mining, Text Summarization, Sarcasm Detection, Information Retrieval, Question Answer System	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the Natural Language in the form of Text
CO2 - Determine the Syntactical Structures of Natural Languages
CO3 - Evaluate the Language Model for Natural Languages
CO4 - Apply Information Retrieval Techniques to Build Search Engines
CO5 - Develop POS tagger, Parsers and Shallow Parser for Natural Languages
CO6 - Design Text Summarization, Sentiment Analysis, Sarcasm Detection, Spelling Correction Systems

TEXT/REFERENCE BOOKS

1. Yoav Goldberg Neural Network Methods for Natural Language Processing, Morgan and Claypool
2. Dan Jurafsky and James Martin Speech and Language Processing.
3. Christopher D. Manning and Hinrich Schütze, Foundations of Statistical Natural Language Processing.
4. James Allen, Natural Language Understanding, Benjamin/Cummings.
5. Natural Language Processing: A Paninian Perspective by Akshar Bharati, Vineet Chaitanya and Rajeev Sangal

COURSE CODE					Fundamental Concepts of Speech Processing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITE: Linear Algebra, Machine Learning

COURSE OBJECTIVES

- Describe the physiological and acoustic aspects of speech production.
- Implement various feature extraction techniques and understand their role in speech processing.
- Implement the principles of speech coding and compression.
- Apply language modelling techniques for improving speech recognition accuracy.

UNIT 1: Fundamentals of Digital Signals and Systems Basic concepts and digital representation of digital signals, Sampling, Quantization, Time Domain analysis of Speech Signals, signal segmentation and framing, Frequency Domain analysis, discrete Fourier transform (DFT) and the fast Fourier transform (FFT), digital filter design principles and techniques, Finite impulse response (FIR) filters and Infinite impulse response (IIR) filters.	**12 Hrs.**
UNIT 2: Introduction to Digital Speech Processing Overview of Speech Processing, Human Speech production, Digitization and recording of speech signal, Acoustic Phonetics and Articulatory Phonetics, Different categories speech sounds and Location of sounds in the acoustic waveform and spectrograms, Uniform Tube Modeling of Speech Production.	**10 Hrs.**
UNIT 3: Human Speech Production and Source Filter Model Speech Perception, Pitch and Formant Analysis, Mel-Frequency Cepstral Coefficients (MFCCs), Speech Feature Extraction Techniques, Segmental and Supra Segmental Features of Speech signal, Speech Prosody, Speech Prosody Modeling (Fujisaki Model).	**10 Hrs.**
UNIT 4: Speech Recognition and Applications Introduction to Speech Recognition, Hidden Markov Models (HMMs), Acoustic Modeling in Speech Recognition, Language Modeling for Speech Recognition, Decoding Techniques (Viterbi Algorithm), Text-to-Speech (TTS) Systems, Speaker Identification, Speaker Verification.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand fundamental concepts of speech production, acoustics, and digital signal processing to represent speech signals.

CO2 - Distinguish feature extraction techniques and understand their role in speech processing applications.

CO3 - Illustrate a foundational understanding of Language Model to implement speech recognition system.

CO4 - Implement the principles of speech coding techniques for compression.

CO5 - Demonstrate speech synthesis techniques for text-to-speech systems.

CO6 - Apply speech processing technologies in real-world applications, including Automatic Speech Recognition (ASR) and Natural Language Processing (NLP).

TEXT/REFERENCE BOOKS

1. Lawrence R. Rabiner, Ronald W. Schafer, An Introduction to Digital Speech Processing: Foundations and Trends® in Signal Processing, Now publishers Inc
2. A. Nejat Ince, Digital Speech Processing: Speech Coding, Synthesis and Recognition, The Springer International Series in Engineering and Computer Science.
3. Branko Kovacevic, Milan M. Milosavljevic, Mladen Veinović, Milan Marković, Robust Digital Processing of Speech Signals, Springer International Publishing AG.

Program Elective Course Track 4

Computing Track

UG_1_T					Advanced Data Structures and Algorithms					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- Understand and implement advanced data structures such as AVL trees, B-trees, red-black trees, Fibonacci heaps, and suffix trees, etc.
- Choose appropriate data structures to design algorithms for a specific problem.
- Develop proficiency in designing and analysing algorithms for complex problems
- Gain hands-on experience through projects or case studies that involve implementing and analysing algorithms and data structures in practical scenarios.

UNIT 1: Hashing And Skip Lists Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists	**10 Hrs.**
UNIT 2: Advanced Trees Revision to BST and AVL trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees, Priority Search Tree, Priority Range Trees, Quad trees, k-D Trees, Fibonacci heaps and suffix trees.	**10 Hrs.**
UNIT 3: Advanced Algorithms Maximum matching in graphs, Edmond's Blossom algorithm for augmenting path, Maxflow-mincut theorem, Ford-Fulkerson and Edmond-Karp maximum-flow algorithm, Chinese Remainder Theorem, DFT and FFT algorithms, Simplex method.	**12 Hrs.**
UNIT 4: Approximation And Randomized Algorithms Interior point method, ellipsoid method, sequential quadratic programming, Randomized Quicksort, Ant colony optimization, Monte-Carlo Method.	**10 Hrs.**
	42 Hrs

COURSE OUTCOMES

After completion of the course, students will be able to:
CO1 - Describe various collision resolution techniques in hashing.
CO2 - Understand properties and operations associated with advanced tree data structures.
CO3 - Apply advanced trees to design efficient data structures for specific applications, such as priority search trees or suffix trees.
CO4 - Analyse the time and space complexities of standard advanced algorithms.
CO5 - Evaluate the suitability of different approximation and randomized algorithms for solving specific optimization problems.
CO6 - Develop novel algorithms or variations of existing algorithms to address optimization problems, considering both deterministic and randomized approaches.

TEXT/REFERENCE BOOKS

1. Mark Allen Weiss - Data Structures and Algorithm Analysis in C++, Pearson.
2. M T Goodrich, Roberto Tamassia - Algorithm Design, John Wiley.
3. Charles E. Leiserson, Thomas H. Cormen, Ronald L. Rivest, Clifford Stein - Introduction to Algorithms, MIT Press.
4. Klienberg and Tardos, Algorithm Design, Pearson.

Course Code					Blockchain Technology					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the concepts of distributed consensus and trust management.
- To understand the design principles of the block chains.
- To design and implement the distributed ledger and the smart contracts.

UNIT 1: Blockchain Fundamentals Basic Data Structure concepts related to blockchain. Introduction to Blockchain, Cryptographic primitives used in Blockchain. Basic Distributed System concepts – distributed consensus and Byzantine fault-tolerant consensus methods.	**11 Hrs.**
UNIT 2: Blockchain Types And Cryptocurrency Proof-of-Work based consensus mechanisms, Proof of Stake based Chains, Types of Blockchain. Introduction to Crypto Currency, Crypto Currency as application of blockchain technology.	**11 Hrs.**
UNIT 3: Smart Contract And Blockchain Platform Blockchains with smart contracts - Ethereum platform and its smart contract mechanism.	**10 Hrs.**
UNIT 4: Blockchain Use Cases Blockchain Use Cases – Finance, Industry, E-Governance and other contract enforcement mechanisms. Security and Research Aspects in Blockchain.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

At the end of the course, students will be able to -

CO1 - Define the role of Blockchain technology in digitization.
CO2 - Illustrate the cryptographic concepts, and distributed concepts related to Blockchain technology.
CO3 - Experiment with Ethereum framework for Blockchain development.
CO4 - Analyze the need of Blockchain for real life system.
CO5 - Choose the appropriate type of Blockchain, and framework according to Blockchain usecase.
CO6 - Create the smart contracts and Blockchain for suitable system.

TEXT/REFERENCE BOOKS

1. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guide to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform
2. Mark Gates, "Blockchain: Ultimate guide to understanding blockchain, bitcoin, cryptocurrencies, smart contracts and the future of money", WiseFox publishing.
3. Debajani Mohanty, "Ethereum for Architects and Developers: With Case Studies and Code Samples in Solidity", Apress.

Course Code					Big Data Analytics					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Learn the issues of Big Data.
- Learn to build and maintain reliable, scalable, distributed systems with Hadoop.
- Able to apply Hadoop ecosystem components.
- Working with SPARK for Data Analysis.

UNIT 1: Introduction Introduction to Big Data, different Vs in Big Data Analytics, Data Serialization, Apache Hadoop & Hadoop Ecosystem, Introduction to MAP-Reduce Architecture, Analysing Data with Hadoop, Hadoop Streaming	**11 Hrs.**
UNIT 2: Distributed File System Distributed File System, the Design of HDFS, HDFS Concepts, Interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O, Avro and File-Based Data structures. Introduction to Scala programming and platforms.	**10 Hrs.**
UNIT 3: Processing Of Big Data Anatomy of a Map Reduce Job Run, Map Reduce Types and Formats, Map Reduce Features, Failures, Job Scheduling, Shuffle and Sort, Matrix-Vector multiplication by Map-reduce, Task Execution, Introduction to PIG, HIVE, Kafka.	**11 Hrs.**
UNIT 4: Data Analysis With Spark SPARK Introduction to Data Analysis with Spark, Downloading Spark and Getting Started, Programming with RDDs, Machine Learning with MLlib, Introduction to NoSQL tools, Data Visualization tools, Analytics using R programming.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the fundamental concepts of Big Data management and analytics
CO2 - Learn the concepts and methodologies for Map Reduce Programming
CO3 - Evaluate tools of Hadoop echo system
CO4 - Identify and exploit different features of SPARK for big data applications
CO5 - Apply the skills to conduct different types of data analysis
CO6 - Develop effective strategies for predictive and streaming data analysis for real world applications

TEXT/REFERENCE BOOKS

1. Chris Eaton, Dirk deroos et al., “Understanding Big Data”, McGraw Hill.
2. Tom White, “HADOOP: The definitive Guide”, O Reilly.
3. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley.
4. Donald Miner & Adam Shook, “MapReduce Design Patterns”
5. Bill Chambers, Matei Zaharia, “Spark: The Definitive Guide”, O'Reilly Media, Inc

Cloud Computing										
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/ Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To provide an in-depth and comprehensive knowledge of Cloud Computing fundamental issues, technologies, applications and implementations.
- To expose the students to the frontier areas of Cloud Computing
- To motivate students to use various cloud computing environments in their future projects.
- To shed light on the issues and challenges in Cloud Computing

UNIT 1: Cloud Foundation And Overview Distributed Computing, Cluster computing, Grid computing. Cloud Service Models	**11 Hrs.**
UNIT 2: Virtualization And Load Balancing Virtualization concepts - Types of Virtualization, Introduction to Various Hypervisors, Moving VMs, Pros and cons of virtualization, Virtualization Technology examples. Distributed Management of Virtual Infrastructures, Scheduling, Capacity Management to meet SLA Requirements, Various load balancing techniques.	**11 Hrs.**
UNIT 3: Industrial Platforms And New Developments Study of Cloud Computing Systems like Amazon EC2 and S3, Google App Engine, and Microsoft Azure, Build Private/Hybrid Cloud using open-source tools. MapReduce and its extensions to Cloud Computing, Cloud Application Programming, and the Aneka Platform	**11 Hrs.**
UNIT 4: Advanced Topics In Cloud Computing Energy efficiency in clouds, Market-based management of clouds, Federated clouds, Security in Cloud Computing	**9 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Explain the strengths and limitations of cloud computing and the possible applications for state-of-the-art cloud computing.

CO2 - Identify the architecture and infrastructure of cloud computing.

CO3 - Demonstrate the concept and role of virtualization in cloud computing.

CO4 - Provide the appropriate cloud computing solutions and recommendations as per the applications.

CO5 - Understand the tradeoffs in power, efficiency, and cost in cloud computing.

CO6 - Analyze authentication, confidentiality and privacy issues in cloud computing

TEXT/REFERENCE BOOKS

1. Rajkumar Buyya, James Broberg, Andrzej M Goscinski, Cloud Computing: Principles and Paradigms, Wiley publication
2. Toby Velte, Anthony Velte, Cloud Computing: A Practical Approach, McGraw-Hill Osborne Media
3. K. Chandrasekaran, Essentials of Cloud Computing
4. Recent publications for case studies

Course Code					Quantum Computing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- Learn the basics of quantum computing.
- Learn important applications of quantum information processing.
- Understand the various cryptography techniques and algorithms that use quantum computing.
- Know about the physical realization of a quantum computer and get basic knowledge of the development libraries for quantum computer programming.

UNIT 1: Introduction To Quantum Computing Introduction, quantum bits, multiple qubits, quantum gates, quantum circuits, no cloning theorem, linear algebra and postulates for quantum mechanics, density operator, Pauli operators, Bloch sphere representation.	**12 Hrs.**
UNIT 2: Quantum Information Processing Bell states, quantum entanglement, EPR inequality, quantum teleportation, quantum super dense coding, pure and mixed quantum states, quantum fidelity.	**12 Hrs.**
UNIT 3: Quantum Cryptography And Quantum Algorithms BB84 protocol, B91 protocol, Ekert protocol, quantum secret sharing, Quantum parallelism, Deutsch algorithm, Deutsch-Josza and Bernstein Vazirani Algorithms, Simon Problem, Grover's Search algorithm, Quantum Fourier Transform, Shor's Factorization Algorithm, Quantum noise and error correction.	**10 Hrs.**
UNIT 4: Quantum Computers: Physical Realization And Development Libraries Stern Gerlach experiment, Quantum computers using ion traps, superconducting circuits, and nuclear magnetic resonance, development libraries for quantum computer programming, quantum computing simulators like IBM Quantum experience, Qiskit and Quantum Development Kit.	**8 Hrs.**
	42 Hrs.

COURSE OUTCOMES

At the end of the course, students will be able to -

CO1 - Illustrate the potential of quantum computing.
CO2 - Understand qubits, entanglement, quantum gates, and their representation.
CO3 - Analyze the applications of quantum information processing, like teleportation and super dense coding.
CO4 - Describe quantum cryptography techniques.
CO5 - Compare quantum algorithms with their classical counterparts.
CO6 - Simulate basic quantum circuits on IBMQ.

TEXT/REFERENCE BOOKS

1. Michael A. Nielsen and Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press.
2. Chris Bernhardt, Quantum Computing for Everyone, MIT Press.
3. Jack Hidary, Quantum Computing: An Applied Approach, Springer.
4. Robert S. Sutar, Dancing with Qubits: How quantum computing works and how it can change the world, Packt Publishing Ltd..
5. Vishal Sahni, Quantum Computing, TATA McGrawHill.

Program Elective Course Track 5

Communication Track

UG_1_T					Data Communication					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the importance of data Communication.
- To understand the fundamental communication concepts of digital and analog transmissions.
- To demonstrate digital signal transmission and encoding techniques.
- To illustrate the working of signal and carrier wave multiplexing techniques.

UNIT 1: Data Communication Overview and Physical Layer Introduction – Process of Data Communication, Transmitter, Receiver, Medium and Message Protocol, Protocols, Standards and Standard organizations, Bandwidth, Data Transmission Rate, Baud rate and Bits per second, modes of communication, OSI/TCIP Model, Networks, Internet, Intranet, Protocols, Telephone and Cable network, Physical Layer – Signals, Analog, Digital, Analog vs. Digital, Transmission impairment, Data Rate Limits, Noiseless and Noisy channel, Performance	**11 Hrs.**
UNIT 2: Data Transmission and Signal Encoding Techniques Analog Transmission – Analog to Analog conversion: Amplitude Modulation, Frequency Modulation, and Phase Modulation, Digital to Analog conversion: ASK, FSK, PSK, and QAM. Digital Transmission – Analog to digital conversion, Digitization, PCM, Transmission Modes, Line Coding (Unipolar, Polar, Biphase), Block Coding (4B/5B Encoding), DTE-DCE Interface, Modems, 56K Modems, Cable Modems.	**12 Hrs.**
UNIT 3: Transmission Media Transmission Media – Guided Media: Twisted Pair cable, Coaxial cable, and Fiber optic cable, Unguided Media: Radio Waves, Microwaves, and Infrared satellite, Line-of-site transmission: point-to-point and broadcast.	**08 Hrs.**
UNIT 4: Multiplexing Multiplexing – FDM, WDM, Synchronous TDM (time slots & frames, interleaving, data rate management), Statistical Time Division Multiplexing, Spread Spectrum – FHSS, DSSS, Switching: Circuit-Switched Networks, Datagram networks, Concept of Virtual circuit networks, structure of circuit and packet switch. Concepts of DSL and ADSL.	**11 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, students will be able to

CO1 - Understand the fundamental concepts of Data Communication and its components.
CO2 - Illustrate the digital and analog modes of transmission and its conversion.
CO3 - Discuss and select appropriate signal encoding technique for transmission.
CO4 - Analyze the data communication and carrier equipment.
CO5 - Compare transmission media based on transmission impairments and channel capacity.
CO6 - Examine multiplexing and switching techniques for signal transmissions.

TEXT/REFERENCE BOOKS

1. Data Communications and Networking, Behrouz Forouzan, McGraw-Hill.
2. Data and Computer Communications, William Stallings, Prentice Hall.
3. Computer Networks, Andrew Tanenbaum, Prentice Hall.

UG_1_T					Mobile Computing					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

COURSE OBJECTIVES

- To understand the basic concepts of mobile computing.
- To study the various generation of the mobile communication system.
- To demonstrate the various cellular networking architecture.
- To illustrate the working of various routing protocols in mobile networks.

UNIT 1: Introduction Introduction to Mobile Computing, Applications of Mobile Computing, Generations of Mobile Communication Technologies and their technical specification, Multiplexing -MAC Protocols SDMA- TDMA- FDMA- CDMA	**10 Hrs.**
UNIT 2: Cellular Mobile Wireless System Introduction to Cellular Systems, GSM- Multiple Access Scheme, GSM Protocols and Signalling, Connection Establishment Frequency Allocation, Routing, Mobility Management, GPRS- Protocols, Network Architecture, UMTS- Network Protocol Architecture, Mobility Management.	**12 Hrs.**
UNIT 3: Mobile Ad-Hoc Network Mobile IP- Operations, Registration, Tunnelling, Proactive protocol- Destination-Sequenced Distance-Vector Routing, Reactive Routing Protocols- Dynamic Source Routing, Ad hoc On-Demand Distance Vector, Hybrid routing- Zone Routing Protocol, Multicast Routing- On-Demand Multicast Routing Protocol, Vehicular Ad Hoc Networks, MANET Vs VANET	**10 Hrs.**
UNIT 4: Mobile Transport and Application Layer Mobile TCP, Wireless Application Protocol, Wireless Transport Layer Security, Wireless Transaction Protocol, Wireless Application Environment, Wireless Telephony Application.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the fundamental concepts and principles of mobile computing.
CO2 - Illustrate the GSM, GPRS and UMTS cellular networking architecture.
CO3 - Discuss the mobility management in the cellular networks.
CO4 - Analyze the proactive and reactive routing strategies in the Ad-hoc network.
CO5 - Compare the Vehicular Ad-hoc Networks with Mobile Ad-hoc Networks.
CO6 - Determine the functionality of transport and application layer protocols.

TEXT/REFERENCE BOOKS

1. Jochen Schiller – Mobile Communication –PHI
2. Prasant Kumar Pattnaik, Rajib Mall- Fundamentals of Mobile Computing, PHI
3. A.Goldsmith, Wireless Communications, Cambridge University Press.
4. Siva Ram Murthy C and Manoj B S, “Ad Hoc Wireless Networks: Architectures and Protocols”, Prentice Hall.

UG_1_T					Wireless Sensor Networks					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

PREREQUISITE: Computer Networks

COURSE OBJECTIVES

- To understand the concepts of sensor networks.
- To study the various protocols at various layers and its differences with traditional protocols.
- To understand the applications and security of Ad-Hoc and sensor networks.

UNIT 1: INTRODUCTION Key definitions of sensor networks, Advantages of sensor networks, Unique constraints and challenges, Driving applications, Enabling technologies for wireless sensor networks, Single-Node architecture, Hardware components, Energy consumption of sensor nodes, Network architecture -sensor network scenarios, Optimization goals and Figures of merit.	**10 Hrs.**
UNIT 2: MAC PROTOCOLS Introduction, Classifications of MAC protocols, Contention-based protocols, Contention-based protocols with reservation mechanisms, Contention-based MAC protocols with scheduling mechanisms, Other MAC protocols.	**11 Hrs.**
UNIT 3: ROUTING PROTOCOLS Introduction, Classification of routing protocols, Table-driven routing protocols, On- demand routing protocols, Hybrid routing protocols, Routing protocols with efficient flooding mechanisms, Hierarchical routing protocols, Power-aware routing protocols, Proactive routing.	**11 Hrs.**
UNIT 4: TRANSPORT LAYER AND SECURITY CHALLENGES Introduction to transport layer protocols, Classification of transport layer solutions, TCP over Ad Hoc wireless networks, Other transport layer protocol for Ad Hoc wireless networks. Security in Ad Hoc wireless networks, Network security requirements, Issues and challenges in security provisioning, Network security attacks, Secure routing in Ad Hoc wireless networks.	**10 Hrs.**
	42 Hrs

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Understand the basics of wireless sensor networks and its applications.
CO2 - Understand typical node and network architectures.
CO3 - Compare different MAC layer protocols of the wireless sensor networks.
CO4 - Analyse routing layer protocols of the wireless sensor networks.
CO5 - Examine transport layer protocols of the wireless sensor networks.
CO6 - Identify and address the security issues in wireless sensor networks.

TEXT/REFERENCE BOOKS

1. C. Siva Ram Murthy and B.S.Manoj. “Ad-Hoc Wireless Networks: Architectures and Protocols”, Pearson Education.
2. Jagannathan Sarangapani. “Wireless Ad-hoc and Sensor Networks: Protocols, Performance and Control”, CRC Press.
3. Holger Karl & Andreas Willig, “Protocols and Architectures for Wireless Sensor Networks”, John Wiley.
4. Kazem Sohraby, Daniel Minoli, & Taieb Znati, “Wireless Sensor Networks- Technology, Protocols, and Applications”, John Wiley.

UG_1_T					Internet of Things					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To impart knowledge about components of IoT Architecture and platforms of IoT.
- To study different types of protocols used in IoT.
- To develop real life IoT based solutions.

UNIT 1: Introduction To Internet Of Things IoT Definition, IoT characteristics, M2M and IoT, End to End IoT Architecture, Physical design of IoT, Logical design of IoT, Challenges for IoT, Interdependencies of IoT and cloud computing, Web of things, IoT data management and compute stack	**10 Hrs.**
UNIT 2: Iot PROTOCOLS Overview of IoT protocols, IoT access technologies: IEEE 802.15.4, ZigBee, LoRaWAN, IP as IoT network layer: 6LoWPAN, 6Lo, RPL, Transport layer protocols, Application layer protocols-CoAP, MQTT.	**12 Hrs.**
UNIT 3: Embedded Iot Devices Sensors and actuators for IoT applications, IoT components, Smart objects, Connecting smart objects: Communication criteria, Basics of sensor networks, Types and configurations of gateways, Specifications of IoT gateways, Digital Twin.	**10 Hrs.**
UNIT 4: Application Building With IOT Security and IoT, Data Analytics for IoT, Big Data analytics tools and technology, Various applications of IoT: Smart city, Smart grid, Smart transportation, Smart manufacturing, Smart healthcare	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1- Understand the Internet of Things and its components.
CO2- Correlate the IoT architecture with various technological resources/ enabling technologies.
CO3- Compare different protocols/communication technologies of the IoT protocol stack.
CO4- Determine I/O device interfaces, sensors, actuators & communication modules for IoT application
CO5- Analyze different security aspects for the Internet of Things.
CO6- Classify IoT applications and design IoT solutions to lead the smart world.

TEXT/REFERENCE BOOKS

1. D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, J. Henry. “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, Cisco Press.
2. A. Bahga, V. Madisetti. “Internet of Things: A Hands-on Approach”, Universities Press (India) Pvt. Ltd.
3. S. Vasudevan, A. Nagrajan and R.M.D. Sundaram. “Internet of Things”, Wiley India
4. Y. Kanetkar, S. Korde. “21 IoT Experiments”, BPB

UG_7_T					Autonomous System					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs/ Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	-	-	100

PREREQUISITE: Internet of Thing, Computer Programming, Artificial Intelligence

COURSE OBJECTIVES

- Learn the basics of autonomous systems and their technologies.
- Apply algorithms to develop autonomous system solutions.
- Integrate data and intelligence to improve traditional practices in autonomous system
- Design intelligent machines for autonomous system applications.

UNIT 1: Introduction to Autonomous Systems Introduction and Evolution of Robotics and Autonomous Systems, Basics of Robotics and Control Systems, Sensors and Actuators in Autonomous Systems, Basics of Navigation and Path Planning, Introduction to Communication Protocols for Autonomous Systems, Introduction to UAVs: Types, Applications, and Challenges, Expert System and Deep Learning in Autonomous Systems	**10 Hrs.**
UNIT 2: Perception and Sensing Overview of Perception in Autonomous Systems, IOT for Autonomous system, Sensor Fusion Techniques, Computer Vision for Object Detection and Recognition, Deep Learning for Perception in Autonomous Systems, UAV Perception Systems: Cameras, Lidar, and Radar, Case Studies and Applications in UAV Perception.	**10 Hrs.**
UNIT 3: Cross-Domain Integration of Autonomous Systems Edge and Fog Computing for Autonomous Systems, Communication Protocols and Networking for Multi-agent Systems, Human-Robot Interaction and Collaboration, Autonomous Systems in Challenging Environments (e.g., Space, Underwater), Advanced Deep Learning Techniques for Autonomous Systems: Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs), etc.	**12 Hrs.**
UNIT 4: Case Studies and Application of Autonomous System Case studies in Transportation: Autonomous Vehicles in Urban and Rural Environments, Healthcare: Robotics and AI in Medical Diagnosis and Treatment, Agriculture: Drones and Robotics for Precision Farming, Environmental Monitoring: UAVs and Robotics for Surveillance and Conservation, Smart Cities: Integration of Autonomous Systems for Urban Development.	**10 Hrs.**
	42 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 - Gain an understanding of the foundational components and key technologies underpinning autonomous systems, including robotics, UAVs, navigation, communication, fog/edge computing, and data analysis.

CO2 - Apply a variety of computational techniques and algorithms to design, develop, and deploy autonomous systems, ensuring proficiency in tasks such as path planning, perception, decision-making, and control.

CO3 - Integrate data from sensors, actuators, and other sources to enable intelligent decision-making and enhance operational efficiency in diverse applications of autonomous systems, thereby transforming traditional business models and processes.

CO4 - Develop intelligent and automated machines for traditional industries, leveraging advancements in robotics, AI, and automation to optimize production processes, improve quality, and reduce human intervention.

CO5 - Utilize data analytics and artificial intelligence to create Smart World solutions, enabling autonomous systems to enhance urban infrastructure, transportation networks, healthcare delivery, environmental monitoring, and other aspects of daily life.

CO6 - Understand the concept, significance, and strategies for achieving sustainable development in the context of autonomous systems, considering ethical, social, and environmental implications to ensure responsible deployment and utilization of these technologies.

TEXT/REFERENCEBOOKS

1. "New autonomous systems." by Cardon, Alain, and Mhamed Itmi. by John Wiley & Sons.
2. "The The Complete Edition–Software Engineering for Real-Time Systems: A software engineering perspective toward designing real-time systems" by Cooling, Jim. Packt Publishing Ltd.
3. "The Autonomous System: A foundational synthesis of the sciences of the mind." by De Gyurky, Szabolcs Michael, and Mark A. Tarbell. John Wiley & Sons.