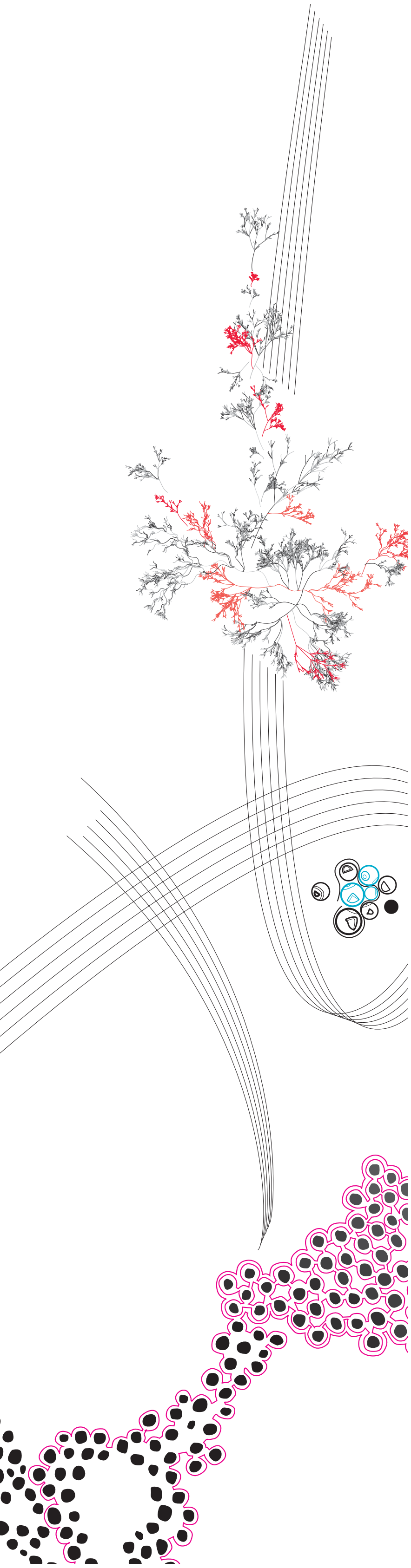

Computer Science and Engineering

B.Tech. Curriculum

2023 Batch Onwards

IIITDM
KURNOOL

Department of Computer Science and Engineering
Indian Institute of Information Technology Design and Manufacturing Kurnool
Jagannathagattu, Andhra Pradesh
India. 518008.

Scheme

First Semester is common to all the branches. The branch change may permitted after the first semester.

3.1 Scheme

First Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS101	Differential and Integral Calculus	BSC	3	0	3
2	DS102	Engineering Physics I	BSC	3	0	3
3	ME101	Engineering Mechanics	BEC	3	0	3
4	EC101	Basic Electrical and Electronics Engineering	BEC	3	0	3
5	CS101	Problem Solving and Computer Programming	BEC	3	0	3
6	DS103	Technical and Professional Communication	HMC	2	2	3
7	DS104	Engineering Physics I Practice	BSC	0	3	2
8	CS102	Problem Solving and Computer Programming Practice	BEC	0	3	2
9	ME102	Engineering Skills Practice	BEC	0	3	2
10	ID101	NSS/NCC/NSO	HMC	0	0	0
Total No. of Credits						24

Note:

- Branch change may be permitted after successful completion of 1st Semester
- NSS/NSO/NCC must be offered as a mandatory course to all the students of 1st Semester. However, students will be given choice to register at least 01 course from NSS, NSO and NCC

Second Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS151	Probability and Statistics	BSC	3	0	3
2	DS152	Linear Algebra	BSC	3	0	3
3	DS153	Modern Physics	BSC	3	0	3
4	DSxxx	Design Elective	DES	2	0	2
5	DSxxx	Humanities Elective	HMC	2	0	0
6	ME151	Concepts in Engineering Design	DES	3	0	3
7	ME152	Design Realization Practice	DES	0	3	2
8	DS154	Modern Physics Practice	BSC	0	3	2
9	CS151	Digital Logic Design	PEC	2	3	4
10	CS152	Python Programming Practice	PEC	0	3	2
11	ID101	NSS/NCC/NSO	HMC	0	0	0
Total No. of Credits						24

Third Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS201	Random Variables and Stochastic Processes	BSC	3	0	3
2	CS201	Discrete Mathematics	PEC	3	0	3
3	CS202	Data Structures	PEC	3	0	3
4	CS203	Computer Organization and Architecture	PEC	3	0	3
5	CS204	Design and Analysis of Algorithms	PEC	3	0	3
6	CS205	Data Structures Practice	PEC	0	3	2
7	CS206	Computer Organization and Architecture Practice	PEC	0	3	2
8	CS207	Design and Analysis of Algorithms Practice	PEC	0	3	2
9	IDxxx	Self Improvement Course - I	HMC	0	0	0
Total No. of Credits						21

Fourth Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS251	Automata Theory and Compiler Design	PEC	3	0	3
2	CS252	Artificial Intelligence	PEC	3	0	3
3	CS253	Operating Systems	PEC	3	0	3
4	CS254	Data Communication and Computer Networking	PEC	3	0	3
5	CS255	Embedded Systems and IoT	PEC	3	2	4
6	CS256	Automata and Compiler Design Practice	PEC	0	3	2
7	CS257	Operating Systems Practice	PEC	0	3	2
8	CS258	Computer Networks Practice	PEC	0	3	2
9	xxxxx	Skill Development Course-I	PDC	2	0	2
10	IDxxx	Self Improvement Course - II	HMC	0	0	0
Total No. of Credits						24

Fifth Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS301	Managerial Economics and Financial Accountancy	HMC	3	0	3
2	CS301	Machine Learning	PEC	3	0	3
3	CS302	Database Management Systems	PEC	3	0	3
4	CS303	Software Engineering	PEC	3	0	3
5	CS304	Data Mining	PEC	3	0	3
6	CS305	Machine Learning Practice	PEC	0	3	2
7	CS306	Database Management Systems Practice	PEC	0	3	2
8	CS307	Software Engineering Practice	PEC	0	3	2
9	IDxxx	Self Improvement Course - III	HMC	0	0	0
Total No. of Credits						21

Sixth Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS351	Entrepreneurship and Management Functions	HMC	3	0	3
2	CS351	High Performance and GPU Computing	PEC	3	0	3
3	CS352	Computer Vision	PEC	3	0	3
4	CS4xx	Department Elective - I	PEC	3	0	3
5	xxx	Free Elective - I	PEC	3	0	3
6	CS353	HPC and GPU Computing Practice	PEC	0	3	2
7	CS354	Computer Vision Practice	PEC	0	3	2
8	ID351	Product Design Practice	DES	0	3	2
9	xxxxx	Skill Development Course-II	PDC	2	0	2
10	IDxxx	Self Improvement Course - IV	HMC	0	0	0
Total No. of Credits						23

Seventh Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS498	Project Phase - I	PDC	0	0	3
2	CS4xx	Department Elective - II	PEC	3	0	3
3	CS4xx	Department Elective - III	PEC	3	0	3
4	CS4xx	Department Elective - IV	PEC	3	0	3
5	XXxxx	Free Elective - II	PEC	3	0	3
6	CS496	Seminar	PCD	0	2	1
Total No. of Credits						16

Eighth Semester						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS499	Project Phase - II	PCD	0	0	8
2	CS497	Internship	PCD	0	0	3
Total No. of Credits						11

3.2 Design Electives

List of Design electives						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS131	Design History	DES	2	0	2
2	DS132	Sociology of Design	DES	2	0	2
3	DS133	Earth, Environment and Design	DES	2	0	2

3.3 Humanities Electives

List of Humanities Courses						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	DS136	Professional Ethics for Engineers	HMC	2	0	0
2	DS137	Universal Human Values	HMC	2	0	0
3	DS138	Constitution of India	HMC	2	0	0
4	DS139	History of Science and Technology in India	HMC	2	0	0

3.4 Professional Engineering Courses

List of Professional Engineering Courses						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS401	Game Theory	PEC	3	0	3
2	CS402	Internet of Things	PEC	3	0	3
3	CS403	Natural Language Processing	PEC	3	0	3
4	CS404	Information Retrieval	PEC	3	0	3
5	CS405	Recommender Systems	PEC	3	0	3
6	CS406	Blockchain	PEC	3	0	3
7	CS407	Object Oriented Programming	PEC	3	0	3
8	CS408	Systems Engineering	PEC	3	0	3
9	CS409	Industry 4.0	PEC	3	0	3
10	CS410	Health Analytics	PEC	3	0	3
11	CS411	Social Network Analysis	PEC	3	0	3
12	CS412	Information Theory and Coding	PEC	3	0	3
13	CS413	Unmanned Aerial Vehicle	PEC	3	0	3
14	CS414	Robotics	PEC	3	0	3
15	CS415	Time Series Analysis	PEC	3	0	3
16	CS416	Information Security	PEC	3	0	3
17	CS417	Human Computer Interaction	PEC	3	0	3
18	CS418	Design Patterns	PEC	3	0	3
19	CS419	Modern Computational Algebra	PEC	3	0	3
20	CS420	Affective Computing	PEC	3	0	3
21	CS421	Genetic Algorithms	PEC	3	0	3
22	CS422	Simulation and Modelling	PEC	3	0	3
23	CS423	Decision Science	PEC	3	0	3
24	CS424	Quantum Computing	PEC	3	0	3
25	CS425	Software Testing	PEC	3	0	3
26	CS426	Soft Computing	PEC	3	0	3
27	CS427	Data Science	PEC	3	0	3
28	CS428	Advanced Data Structures and Algorithms	PEC	3	0	3
29	CS429	Graph Theory	PEC	3	0	3
30	CS430	Approximation and Randomized Algorithms	PEC	3	0	3
31	CS431	Digital Image Processing	PEC	3	0	3
32	CS432	Mobile Computing	PEC	3	0	3
33	CS433	Pattern Recognition	PEC	3	0	3
34	CS434	Cloud Computing	PEC	3	0	3
35	CS435	Parallel and Distributed Systems	PEC	3	0	3
36	CS436	Business Intelligence	PEC	3	0	3
37	CS437	Service Oriented Architecture	PEC	3	0	3
38	CS438	Real-time Systems	PEC	3	0	3
39	CS439	Big Data Analytics	PEC	3	0	3
40	CS440	Fault-Tolerant Systems	PEC	3	0	3

List of Professional Engineering Courses						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
41	CS441	Computer Graphics and Multimedia	PEC	3	0	3
42	CS442	Scalable Data Science	PEC	3	0	3
43	CS443	Optimization Techniques	PEC	3	0	3
44	CS444	Embedded Systems	PEC	3	0	3
45	CS445	Data Privacy	PEC	3	0	3
46	CS446	Marketing Science and Predictive Analysis	PEC	3	0	3
47	CS447	Cyber Forensics	PEC	3	0	3
48	CS448	Statistical Learning	PEC	3	0	3
49	CS449	Financial Analytics	PEC	3	0	3
50	CS450	Data Analysis and Visualization	PEC	3	0	3
51	CS451	Tools and Data Management	PEC	3	0	3
52	CS452	Decision Support Systems	PEC	3	0	3
53	CS453	Immersive Technologies	PEC	3	0	3
54	CS454	Bio-Inspired Computing	PEC	3	0	3
55	CS455	Bioinformatics	PEC	3	0	3
56	CS456	Advanced Data Mining	PEC	3	0	3
57	CS457	Computational Learning Theory	PEC	3	0	3
58	CS458	Algorithmic Techniques for Big Data	PEC	3	0	3
59	CS459	Mining of Massive Datasets	PEC	3	0	3
60	CS460	Virtual Reality and Augmented Reality	PEC	3	0	3
61	CS461	Cluster Computing	PEC	3	0	3
62	CS462	Wireless Networked Systems	PEC	3	0	3
63	CS463	Post Quantum Cryptography	PEC	3	0	3
64	CS465	Web Technologies	PEC	3	0	3
65	CS466	Cryptography and Network Security	PES	3	0	3
66	CS467	Principles of Programming	PES	3	0	3
67	CS468	Remote Sensing and GIS	PES	3	0	3
68	CS469	Gen AI	3	0	3	

3.6 Self Improvement Courses

List of Self Improvement Courses						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	ID401	Physical (Sports, Dance, Body Building, etc.)	HMC	0	0	0
2	ID402	Health and Nutrition (Medical) Awareness (Basics awareness on Medicine, Diseases, Remedie, Nutrition, etc)	HMC	0	0	0
3	ID403	Personality Enhancement/Character Building (emotional development, Reading biography of great personalities, Responsibility and ownership towards nation,)	HMC	0	0	0
4	ID404	Language and music (Learning new languages, poetry, public speaking, vocabulary enhancement, singing, musical instrument, etc)	HMC	0	0	0
5	ID405	Passion (Any passion of the student not registered earlier)	HMC	0	0	0
6	ID406	Financial Awareness (Awareness on Insurances, Investments, financial planning, Money Management, etc)	HMC	0	0	0

3.5 Free Electives

List of Open Electives						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS470	Principles of Programming	PEC	3	0	3
2	CS471	Artificial Intelligence and Machine Learning	PEC	3	0	3
3	CS472	Introduction to Data Science	PEC	3	0	3
4	CS473	Systems Engineering	PEC	3	0	3
5	DS470	Differential Equations	PEC	3	0	3
6	DS471	Materials and Devices for Quantum Technologies	PEC	3	0	3
7	DS472	Numerical Methods	PEC	3	0	3
8	DS473	Quantum Computing	PEC	3	0	3
9	DS474	Sustainable Design	PEC	3	0	3

3.7 Skill Development Courses

Every student has a unique skill and it is essential to rightly identify at the right time and nurture it in the right way. The department took initiatives to encourage the students to undergo two skill development trainings. Technology keeps updating and we are at a compulsion of updating ourselves with the latest skills and it holds good for every profession. Skill development classifies into two types – Soft Skill and Hard Skill development. Soft skills are generally related to personality development and Hard skill is aligned with technical expertise and is oriented with subject knowledge. The scope of Skill development course mainly aims to introduce market-relevant skills. The focus will be to provide essential training and adapt to the latest technology. These programs will pave way for young talent to grab the industry opportunities which will, in turn, improve the underdeveloped sectors.

List of Skill Development Courses						
Sl. No.	Course Code	Course Name	Category Code	I	P	Credits
1	CS381	Linux Administration	PCD	2	0	2
2	CS382	Full Stack Development	PCD	2	0	2
3	CS383	DevOps Toolchain	PCD	2	0	2
4	CS384	Digital Marketing	PCD	2	0	2
5	CS385	Graphic Design and Editing	PCD	2	0	2
6	CS386	Product Automation	PCD	2	0	2
7	CS387	Systems Programming	PCD	2	0	2
8	CS388	Software Toolbox	PCD	2	0	2
9	CS389	Mobile app Development	PCD	2	0	2
10	CS390	Ethical Hacking	PCD	2	0	2
11	CS391	Low Code Development	PCD	2	0	2
12	ME381	Data Science for Engineers	PCD	2	0	2
133	ME382	Basics of Finite Element Analysis	PCD	2	0	2
14	ME383	Fuzzy Logic and Neural Network	PCD	2	0	2
15	ME384	Design, Technology and Innovation	PCD	2	0	2
16	ME385	Manufacturing Guidelines for Product Design	PCD	2	0	2
17	ME386	Artificial Intelligence for Mechanical Engineers	PCD	2	0	2
18	ME387	Database Management System	PCD	2	0	2
19	ME388	Failure Analysis and prevention	PCD	2	0	2
20	EC381	Industry 4.0	PCD	2	0	2

Detailed Syllabi

Each course has a course id, course title and Instructions hours, Practice hour and total Credits (IPC), the type of course namely Core or Elective

Semester I

DS101	Differential and Integral Calculus	3-0-3	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To understand the concept of limits and continuity of functions defined on intervals and apply it to analyze the behavior of functions near a given point. 2. To apply the differentiability concepts and the various Mean Value Theorems to determine the existence of critical points and the behavior of functions in a given interval. 3. To understand the notion of partial derivatives and gradient vectors for functions of several variables, and apply the Chain Rule and Lagrange's Multipliers Method to solve optimization problems. 4. To understand the concept of integrals, the fundamental theorem of calculus, and its applications in evaluating areas, volumes, and solving differential equations. 5. To apply the convergence and divergence tests to determine the convergence of series, understand the difference between absolute and conditional convergence, and apply them in the solution of real-world problems.		
Course Outcomes	At the end of the course, student will be able to: 1. Apply limit concept to determine a given function of one variable is continuous/differentiable at a given point. 2. Find approximation of nonlinear function about a given point using Taylor's formula. 3. Able to determine a given real valued function of two variables is continuous/differentiable at a given point. 4. Find constrained extremum of a given function of several variables by using Lagrange's Method of Multipliers 5. Evaluate given line and surface/multiple integrals. 6. Determine convergence/divergence of a given sequence and series of real numbers.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	1		1		
CO2	3	3	2	2			
CO3	3	3	1				
CO4	3	2	2				
CO5	3	3	3		1		
CO5	3	1	1				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Differential Calculus: Limit and Continuity of functions defined on intervals, Intermediate Value theorem, Differentiability, Rolle's Mean Value Theorem, Lagrange's Mean Value Theorem, Cauchy's Mean Value Theorem, Indeterminate Forms, Taylor's Theorem and Taylor's series.

Functions of Several Variables: Limit and Continuity, Geometric representation of partial and total increments, Partial derivatives, Derivatives of composite and implicit functions (Chain Rule), Change of Variables (Jacobian), Gradient, Lagrange's Multipliers Method for Maxima-Minima, Directional derivatives, Gradient vectors, Tangent plane.

Integral Calculus: Definite integral as the limit of sum, Mean value theorem of integrals, Fundamental theorem of integral calculus and its applications, Differentiation under Integral Sign. Multiple Integrals: Evaluation of double integral in cartesian and polar coordinates, Change of order of integration, Change of Variables, Line integrals, Surface integrals.

Sequences and Series: Sequences of real numbers, Convergence of sequences, Subsequences, Monotone Sequences and their convergence, Cauchy Sequence, Cauchy criterion for convergence of a Sequence, Series of real numbers, Convergent of a Series, Comparison test, Limit Comparison test, D'Alemberts test (Ratio test), Raabe's test, Root test, Integral test for the convergence of series, Alternating series, Leibnitz theorem, absolute and conditional convergence.

Text books:

1. Thomas. G.B, and Finney R.L, Calculus, Pearson Education, 2007.

Reference books:

1. Piskunov. N, Differential and Integral Calculus, Vol. I II, Mir. Publishers, 1981.
2. Kreyszig. E, Advanced Engineering Mathematics, Wiley Eastern 2007.
3. J Hass, M D Weir, F R Giordano, Thomas Calculus, 11th Edition, Pearson.

Others:

1. https://archive.nptel.ac.in/courses/111/106/111106146/

DS102	Engineering Physics	3-0-3	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To understand the fundamental concepts of electrostatics 2. To understand the electrostatic fields 3. To analyze conductors and induced charges, Laplace's and Poisson's equation for electrostatics, uniqueness theorems, the Method of Images, Multipole expansion, and numerical methods to solve Laplace's and Poisson's equations. 4. To understand materials in electric fields, including conductors and dielectrics 5. To understand the concepts of magnetostatics, including the Lorentz Force Law		
Course Outcomes	At the end of the course students will be able to 1. Apply vector calculus for solving problems involving electrostatics. 2. To calculate electric field and potential for various continuous and discrete charge distributions. 3. Apply Biot Savart law to calculate Magnetic Induction. 4. To understand magnetic fields in matter and physics behind different types of magnetic materials. 5. To communicate physics concepts and their applications to engineering fields effectively.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	1	1		2	
CO2	3	2	1	1			
CO3	3	2	1	1			
CO4	3	2	1	1			
CO5	3	2	1	2			

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Electrostatics: Brief review of vectors, Gradient of a scalar field, divergence of a vector, Unit vectors in spherical and cylindrical polar coordinates, Gauss's theorem, Stokes's theorem, The Electric Field, Divergence and curl of electrostatic fields, potential and its relation with electrostatic Field, The energy of a continuous charge distribution, conductors and induced charges, Laplace's and Poisson's Equation for electrostatics, uniqueness theorems, The Method of Images, Multipole expansion, numerical methods to solve Laplace's and Poisson's Equation.

Electric Fields in Matter: Materials in electric field- Conductors-Dielectrics, the field of a polarized object, bound charges, Electric Displacement-Gauss's Law in the Presence of Dielectrics, Susceptibility, Permittivity, dielectric constant, Dielectric strength, Electric flux density, Energy in Dielectric Systems.

Magnetostatics: The Lorentz Force Law, Continuity equation, The Biot-Savart Law, The Divergence and Curl of magnetic field, Magnetic Induction due to configurations of current carrying conductors, Magnetic Vector Potentials, Ampere's Circuit law, Energy density in a Magnetic field.

Magnetic Field in Matter: Magnetic Properties of Materials- Magnetic Dipole, Torques and Forces, Magnetic Permeability and Susceptibility, Magnetic Circuit-Forces in Magnetic Field, Magnetization Dia, Para, ferromagnetism, Effect of a magnetic field on atomic orbits, Bound Currents, Hysteresis loop in ferromagnetic materials.

Text books:

1. W. H. Hayt and J. A. Buck, Engineering Electromagnetics, Tata McFraw Hill Education Pvt. Ltd, 2006.

Reference books:

1. Grifiths. D. J, Introduction to Electrodynamics, Prentice Hall, 2007.

2. Purcell. E.M, Electricity and Magnetism Berkley Physics Course, V2, Tata McGraw Hill, 20 08.

3. Feynman. R.P, Leighton. R.B, Sands. M, The Feynman Lectures on Physics, Narosa Publish ing House, Vol. II, 2008. Hill, 2008.

4. G. B. Arfken, H. J. Weber and F. E. Harris, Mathematical Methods for Physicists, Academic Press, 2013.

Others:

1.https://archive.nptel.ac.in/courses/122/107/122107035/

ME101	Engineering Mechanics 3-0-3 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Understand the fundamental concepts of motion, including vectors, uniform and accelerated motion, and kinematic equations. 2. Develop a deep understanding of Newton's Laws of Motion, including inertia, the invariance of Newton's Laws, and their applications to the dynamics of systems of particles. 3. Explore the principles of conservation of linear and angular momentum, the center of mass, work-energy theorem, potential energy, energy diagrams, and harmonic oscillators. 4. Study the dynamics of rigid bodies, including fixed-axis rotation, Euler's equations, and gyroscope motion and its applications. 5. Analyze the motion under central force, including the law of equal areas, planetary motion, and Kepler's laws.
Course Outcomes	1. Students will be able to apply basic laws and principles of engineering mechanics in real-world situations. 2. Students will be able to draw free body diagrams, study the dynamics of an object and predict the subsequent motion. 3. Students will be able to use polar coordinates to describe rotational motion of an object. 4. Students will be able to understand the planetary motion and gravitation. 5. Students will be able to apply the concepts of angular momentum and torque for rigid body dynamics.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	1	1		2	
CO2	3	2	1	1			
CO3	3	2	1	1			
CO4	3	2	1	1			
CO5	3	2	1	2			

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Vectors, uniform and accelerated motion, kinematic equations, inertia, Newton's Laws, Invariance of newton's laws, relativity. free-body diagrams, equations of motion, some applications of newton's laws.

Dynamics of system of particles, conservation of linear momentum, centre of mass, work energy theorem, potential energy, energy diagrams, Stability analysis, Small Oscillations, harmonic Oscillator. Particle collisions.

Angular momentum and fixed axis rotation, rigid body dynamics, Conservation of angular momentum, Euler's equations, Gyroscope motion and its applications.

Motion under central force, the law of equal areas, planetary motion and Kepler's laws.

Text books:

1. F. Beer. R. Johnston, Vector mechanics for engineers: statics and dynamics. Tata McGraw-Hill, 2010.

Reference books:

1. Meriam. J. L and Kraige. L. G, Engineering Mechanics, Vol. I – Statics, Vol 2: Dynamics, 2007.
2. H. Goldstein, Classical Mechanics, Pearson Education, 2011.
3. Kittle. C, Mechanics – Berkley Physics Course, Vol. 1, Tata McGraw Hill, 2008.
4. Daniel Kleppner and Rober Kolenkow, An introduction to Mechanics, MeGraw Hill Education, 2017.
5. T W B Kibble, Frank H. Berkshire, Classical Mechanics, Imperial College Press, 2004.
6. Somnath Datta, Mechanics, Pearson education India, 2012.
7. A P French, Newtonian Mechanics, Viva Books, 2017.

Others:

1. https://nptel.ac.in/courses/112106286

EC101	**Basic Electrical and Electronics Engineering**	**3-0-3**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. Understand electrical circuit elements and their characteristics, including inductor current and capacitor voltage continuity. 2. Analyze electrical circuits using Kirchhoff's laws and network analysis techniques such as nodal analysis, mesh analysis, and network theorems. 3. Apply network theorems for circuit analysis, including Thevenin's and Norton's theorems, compensation theorem, and maximum power transfer. 4. Analyze RC and RL circuits and measure AC signals, including power factor. 5. Understand the basic characteristics and application of semiconductor diodes, rectifiers, filters, and bipolar junction transistors.		
Course Outcomes	Students will be able to: 1. Analyze and solve electrical and magnetic circuits. 2. Specify the type of electrical machines used for a given application 3. Recognize the ratings of different electrical apparatus 4. Understand the characteristics and applications of semiconductor diodes. 5. Understand the operating principles and DC characteristics of bipolar junction transistors.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	3					
CO2	3	3			2		
CO3	3	2		1			
CO4	3	3	3		2		
CO5	3	2					

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						2	
CO2				1		2	
CO3				1			1
CO4							1
CO5							1

Detailed Syllabus

Electrical circuit elements: voltage and current sources, R,C,L,M,I,V, linear, non linear, active and passive elements, inductor current and capacitor voltage continuity, Kirchhoff's laws, Elements in series and parallel, superposition in linear circuits, controlled sources, energy and power in elements, energy in mutual inductor and constraint on mutual inductance. Network analysis: Nodal analysis with independent and dependent sources, modified nodal analysis, mesh analysis, notion of network graphs, nodes, trees, twigs, links, co-tree, independent sets of branch currents and voltages.

Network theorems: voltage shift theorem, zero current theorem, Tellegen's theorem, reciprocity, substitution theorem, Thevenin's and Norton's theorems, pushing a voltage source through a node, splitting a current source, compensation theorem, maximum power transfer.

RC and RL circuits: natural, step and sinusoidal steady state responses, series and parallel RLC circuits, natural, step and sinusoidal steady state responses. AC signal measures: complex, apparent, active and reactive power, power factor. Introduction to three phase supply: three phase circuits, star-delta transformations, balanced and unbalanced three phase load, power measurement, two wattmeter method. Semiconductor diodes and application: PN diodes, rectifiers and filters, clipping and clamping circuits, voltage multiplier circuits. Bipolar Junction Transistors: DC characteristics, CE, CB, CC configurations, biasing, load line.

Text books:

1. Hayt. W. W, Kemmerly. J.E, and Durbin. S.M, Engineering Circuits Analysis, Tata McGraw Hill, 2008..

2. Boylestad R. Nashelsky L., Electronic Devices Circuit Theory, Pearson Education, 2009.

Reference books:

1. Hughes Edward, Electrical Electronic Technology, Pearson Education, 2007 .

2. Hambley. A, Electrical Engineering Principles and Applications: International Version, Pearson Education, 4 Edn, 2007.

3. Alexander.C. K. Mathew. N. O. Sadiku, Fundamentals of Electrical circuits, Tata McGraw Hill, 2008.

Others:

1. https://archive.nptel.ac.in/courses/108/101/108101091/

CS101	**Problem Solving and Computer Programming**	**3-0-3**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To introduce the basic computer system and command-based computing operations to the students 2. To choose right representation format of data and perform mathematical operations on them programmatically 3. To identify situations where programming can ease the human effort in solving problems 4. To compare different programming constructs and choose the right one based on the requirement of the problem 5. To write, compile, debug a program and finally run it on a computer and to develop logic building skills for solving real-world problems using pseudo code, algorithms and flowcharts		
Course Outcomes	After completion of the course the students should be able to: 1. Define the basic components of a computer system and interpret commands to run and execute programs on Linux based environments 2. Compare and contrast amongst keywords, identifiers, operators, etc. while interpreting the logic through algorithms, flowcharts and pseudo codes 3. Develop programming logic for decision-making through iterative and recursive programming while handling memory allocations through pointers 4. Determine expected output from structured programs while debugging the syntactic and semantic errors 5. Construct programs for problem-solving on real-world data input through console, commands and files		

Problem Solving and Computer Programming

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	2			1	2								2	1
CO2	1	2	1	1									1	2
CO3	1	1	2	3									3	
CO4			1	2	1								2	1
CO5	2	1	3	2									3	2

Detailed Syllabus

History of computers, Basic organisation and building blocks of a digital computer, system and application software, Overview of compiler, linker, loader, interpreter, introduction to flowcharts and algorithms, high level versus machine level instructions, compiling and running C programs through use of standard compilers, basic UNIX commands to run C programs, History of C, Importance of C, basic structure of C program, handling of compilers to develop source codes

Pre-processor directives, macro definitions, use of standard print and scan functions, basic library functions, printing a message, arithmetic operations on inputs supplied by users, interest calculation, ODD-EVEN, printing a sequence of numbers, ASCII operations, Character Set, C Tokens, Keywords and Identifiers, Constants, Variables, Data Types, Declaration of Variables, Declaration of Storage Class, Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operator, Bitwise Operators, Special Operators, Arithmetic Expressions, Evaluation of Expressions, Precedence of Arithmetic Operators

Decision Making with If Statement, Simple If Statement, The If.....Else Statement, Nesting of If....Else Statements, The Else If Ladder, The Switch Statement, The ? : Operator, While Statement, The Do Statement, The For Statement, Goto statement, Jumps in Loops: break and continue, Macro Substitution, File Inclusion, One-Dimensional Arrays, Declaration of One-Dimensional Arrays, Initialization of One-Dimensional Arrays, Array operations: searching, finding min and max, sorting, insertion, deletion, Two-Dimensional Arrays, Initializing Two-Dimensional Arrays, Declaring and Initializing String Variables, Reading Strings from Terminal, Writing Strings to Screen, Arithmetic Operations on Characters, Putting Strings Together, Comparison of Two Strings Pointers and their need, pointer operations, memory allocation in arrays with pointers, operation on arrays using pointers, address of pointers, dynamic allocation of memory, dangling pointers, Need for User-Defined Functions

A Multi-Function Program, Elements of User-Defined Functions, Definition of Functions, Return Values and Their Types, Function Calls, Function Declaration, Passing strings and arrays as parameters to functions, returning pointers using functions, nesting of functions, Recursive functions, difference with the iterative constructs, basic analysis of complexity in the recursive functions
Storage classes in C, scope, visibility, and lifetime of variables, multifile Programs, Basic construct of structures and unions, their necessity, declaring and calling structure variables, difference in memory allocation between structures and unions, array of structures, dynamic array of structures, linking structure variables, dot and indirection operators, Opening a text file in C, modes of access control, reading from and writing to a file, copying the contents, reading special characters, concatenating two files, command-line arguments in C

Text books:

1. Balagurusamy, E. "programming in ANSI C." Tata McGraw-Hill Publishing Company Limited, 2007.
2. Kanetkar, Y.. Let us C. BPB publications, 2018.
3. Deitel H. M. , Deitel P. J.. How to program,, 7th edition, Pearson Education, 2010

Reference books:

1. R. G. Dromey, "How to Solve It By Computer", Pearson, 1982.
2. A.R. Bradley, "Programming for Engineers", Springer, 2011.
3. Kernighan and Ritchie, "The C Programming Language", (2nd ed.) Prentice Hall, 1988.

DS103	Technical and Professional Communication	2-2-3	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. Read a given text at a reasonable speed - Comprehend and critically read the text - Understand and use lexis accurately and appropriately 2. Listen to various types of spoken discourses understand, analyse and apply the same Listen and comprehend lectures and speeches 3. Speak coherently and fluently on a given topic Speak with confidence and present point of view 4. Write fluently and coherently on a given topic - Write various types of tasks short and long 5. Use accurate grammatical structures while speaking and writing		
Course Outcomes	1. Understand basic grammar principles 2. Write clear and coherent passages 3. Write effective letters for job application and complaints 4. Prepare technical reports and interpret graphs 5. Enhance reading comprehension 6. Comprehend English speech sound system, stress and intonation		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO6							

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2	3		1		
CO2	1		3	2	1		
CO3	1	2	3		1		
CO4	1	1	3		1		
CO5		2	3		2		
CO6	1	2	3	2	1		

Detailed Syllabus

Listening – Listening comprehension. Listen to various types of spoken discourses understand, analyse and apply the same. Listen and comprehend lectures and speeches.
Speaking – Organization, articulation and correctness. Speak with confidence and present a point of view. Speak coherently and fluently on a given topic.
Reading – Comprehend and critically read the text. Read a given text at a reasonable speed
Writing – Memos, letters, reports, reviews and writing fluently and coherently on a given topic. Write various types of tasks; short and long.
Presentation Skills – Oral presentation using Power Point. Study Skills – Dictionary, thesaurus reference
Structure of English – Remedial grammar/ Grammar for Communication.

Text books:

1. Shreesh Choudhry, Devaki Reddy, Technical English, Macmillan Publishers,2009.

Reference books:

1. Martin Hewings, Advanced English Grammar, Cambridge University Press,2007.
2. V. Saraswathi, Leena Anil, Manjula Rajan, Grammar for Communication,2012.
3. Thomson and Martinet, Practical English Grammar, Oxford University Press, 1986.
4. Leech, Geoffrey Jan Svartvik, A Communicative Grammar of English, Longman,2003

Others:

1. https://nptel.ac.in/courses/109104031

DS104	Engineering Physics Practice 0-3-2 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To provide students with a practical understanding of engineering physics concepts and principles 2. To develop students' analytical and problem-solving skills through laboratory experiments in physics 3. To enhance students' ability to design, execute, and interpret experimental data
Course Outcomes	1. Develop proficiency in using various equipment and instruments in the physics laboratory 2. Formulate and execute experiments to investigate various laws of physics and verify theoretical concepts 3. Analyze and interpret experimental data to derive meaningful conclusions 4. Demonstrate effective technical communication skills through written reports and oral presentations

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1		1	1				3					
CO2	2	1	1	1	1				3					
CO3	3	1	1	2	1				3					
CO4	3			1	1				3					

List of Experiments

1. Familiarization with laboratory equipment and instruments, introduction to standard laboratory practices and safety guidelines
3. Measurement of length, time, and mass
4. Determination of specific heat capacity and thermal conductivity
5. Verification of laws of motion
6. Study of projectiles and their motion
7. Applications of simple machines
8. Study of wave phenomena and interference
9. Measurement of refractive index of materials
10. Investigation of diffraction and polarization
11. Verification of Ohm's Law
12. Measurement of magnetic fields and forces
13. Investigation of AC and DC circuits

Reference books:

1. Laboratory Manual for Introductory Physics by Michael E. Suydam.
2. Physics Laboratory Experiments by Jerry D. Wilson and Cecilia A. Hernandez-Hall.
3. Principles of Physics Laboratory Manual by Melissinos and Napolitano.
4. Optics and Waves: An Introduction by Bernard H. Brinkman.
5. Fundamentals of Electricity and Magnetism by Arthur F. Kip.

CS102	**Problem Solving and Computer Programming Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To develop programming skills through a first-level course on C programming 2. To introduce basic decision making through programmatic exercises 3. To manifest iterative and recursive paradigms in programming 4. To develop a basic skill-set to work with linux based systems 5. To inculcate the programmatic thinking in the students through coding exercises		
Course Outcomes	1. After completion of this course the students should be able to: 2. Use ubuntu-based computer systems for basic operations 3. Should have the fundamental programming skills required for the advanced courses like data structures, algorithms, operating systems, etc. 4. Appear for different competitive coding exercises to manifest their expertise 5. Should be able to develop programming logic given a real-world problem		

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	2		1	1									2	2
CO2	1	2	2	2									3	2
CO3	2		1											1
CO4	2	2	1	3							1		1	2
CO5		3	2	3	3				2			3	3	3

List of Experiments

1. Draw flowcharts, write algorithms and develop C codes for basic problems like addition of two numbers, displaying a message, basic arithmetic operations and display of results.
2. Basic Ubuntu commands like creating a directory, list all files, copy a file from one place to another, opening a C file, compile and execute it.
3. Write C programs to take input from users, basic decision making, basic iterative constructs, arithmetic operations, multiplication tables, primality tests, etc.
4. Write C programs to use relational, arithmetic, logical, bitwise and ternary operators, etc.
5. Write C programs to solve complicated problems through the use of operators, nested decision making, ternary operators instead of if-else on the same problem, etc.
6. Write C programs to use loops in solving problems, factorial computation, Fibonacci sequences, break and continue constructs, etc.
7. Write C programs for matrix manipulation operations; insertion, display, searching, basic sorting, and deletion exercises on arrays.
8. Write C programs to display, concatenation, selection, capitalisation in strings, palindrome exercises, string comparison, copying strings, etc. without the use of the string library function.
9. Write C programs for simple pointer operations, solution to complicated problems using pointers, dynamic initialisation and array operations using pointers.
10. Write C programs to write functions, using array as arguments to the functions, using pointer as arguments and return-type to the functions.
11. Write C programs on recursive functions, factorial, Fibonacci sequences through recursive constructs, use of global and static variables within the functions, etc.

Additional Experiments: 1. Write C programs to to realise memory allocation in structure versus unions, array of student records in a class, usage of dot and indirection operators, file handling, etc.

Text books:

1. Balagurusamy, E. "programming in ANSI C." Tata McGraw-Hill Publishing Company Limited, 2007.
2. Kanetkar, Y.. Let us C. BPB publications, 2018.
3. Deitel H. M. , Deitel P. J.. How to program,, 7th edition, Pearson Education, 2010

Reference books:

1. R. G. Dromey, "How to Solve It By Computer", Pearson, 1982.
2. A.R. Bradley, "Programming for Engineers", Springer, 2011.
3. Kernighan and Ritchie, "The C Programming Language", (2nd ed.) Prentice Hall, 1988.

1. https://nptel.ac.in/courses/111106146 (Links/Course materials)

ME102	Engineering Skills Practice 0-3-2 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Familiarize students with basic manufacturing processes and electrical wiring. 2. Train students to identify electronic components and assemble simple circuits. 3. Teach students to design and make circuits on PCB, and to solder and test electronic components. 4. Introduce various types of domestic wiring practices and power consumption estimation. 5. Teach students how to identify and assemble computer hardware components and install software.
Course Outcomes	After completion of the course, students can: 1. Work on machine tools and their operations 2. Practice on manufacturing of components using workshop trades including fitting, carpentry, foundry and welding 3. Identify and apply suitable tools for machining processes including turning, facing, thread cutting and tapping 4. Apply basic electrical engineering knowledge for house wiring practice 5. Identify computer peripherals and components, assemble/disassemble hardware, install Windows and Linux OS and open-source software

List of Experiments

Experiments will be framed to train the students in following common engineering practices: Basic manufacturing processes: Fitting – Drilling tapping – Material joining processes – PCB making – Assembling and testing – Electrical wiring.

Familiarization of electronic components by Nomenclature, meters, power supplies, function generators and Oscilloscope – Bread board assembling of simple circuits: IR transmitter and receiver – LED emergency lamp – Communication study: amplitude modulation and demodulation.

PCB: designing and making of simple circuits – Soldering and testing of electronic components and circuits.

Various types of Domestic wiring practice: Fluorescent lamp connection, Staircase wiring – Estimation and costing of domestic and industrial wiring – power consumption by Incandescent, CFL and LED lamps.

Identification of the peripherals of a computer, components in a CPU and its functions, Assembling and disassembling the system hardware components of the personal computer- CPU(Processor), Mother Board, Hard Disk Drive, Monitor, RAM(SD or DDR), Bus Cables, Power Cables, SMPS, etc. Windows, Linux installation, Open source software installations.

Text books:

1. Uppal S. L., "Electrical Wiring Estimating", 5Edn, Khanna Publishers, 2003. .
2. Chapman. W. A. J., Workshop Technology, Part 1 2, Taylor Francis.

Reference books:

1. Clyde F. Coombs, "Printed circuits hand book", 6Edn, McGraw Hill, 2007.
2. John H. Watt, Terrell Croft, "American Electricians' Handbook: A Reference Book for the Practical Electrical Man", Tata McGraw Hill, 2002.

ID101	NSS/NCC/NSO 0-0-0 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To shape the students into strong, disciplined, responsible, and highly motivated citizens 2. Foster in young minds a sense of dedication to the values of national integration, secularism, nationalism, and social welfare 3. To educate the students about physical fitness 4. To emphasises working democratically and uphold the values of unselfish service, respect for another person's viewpoint, and care for fellow humans
Course Outcomes	1. Students develop empathy, and contribute to social causes, leading to a better understanding of societal needs and issues. 2. Students develop leadership skills, communication skills, teamwork, and critical thinking abilities, which are essential for personal growth and career development 3. Students learn about their social responsibility towards nation, and develop a sense of civic duty. 4. Students will be able identify social issues and take proactive measure towards finding solutions. 5. Students develop a sense of responsibility towards the well-being of the community.

Activities:

National Cadet Corps (NCC) conduct atleast 20 parades each year. Conduct social welfare awareness, first aid, and personal hygiene programmes
Conduct sports activities and participate in inter-institutional sports activities
Camps are organised in neighbouring villages where efforts are made to better the people' circumstances. A volunteer with the NSS fills their calendar with events like cleaning up temples, running blood donation camps, and teaching school children how to use computers.

Semester II

DS151	**Probability and Statistics 3-0-3 Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Understand probability theory, including applications in finance, insurance, and engineering. 2. Analyze data using measures of central tendency, dispersion, skewness, kurtosis, and graphs. 3. Develop skills in sampling distributions, estimation, hypothesis testing, and statistical tests. 4. Gain proficiency in statistical software such as R, SAS, and SPSS. 5. Learn advanced statistical techniques such as regression analysis, time series analysis, principal component analysis, factor analysis, cluster analysis, and non-parametric tests.
Course Outcomes	1. Solve problems on discrete and continuous random variables 2. Identify the characteristics of different discrete and continuous distributions. 3. Analyze the statistical problems on large and small samples. 4.Apply the knowledge of probability and statistics in solving engineering problems

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	1	1	1				
CO2	2	2					
CO3	2		1				
CO4	2		1				
CO5	1	2	1				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Introduction to Probability: Sets, Events, Axioms of Probability, Conditional Probability and Independence, Bayes Theorem.

Random Variables: Definitions, Cumulative Distribution Functions, Probability Mass Function, Probability Density Function, Joint and Conditional Distributions.

Expectations: Mean, Variance, Moments, Correlation, Chebychev and Schwarz Inequalities, Moment-Generating and Characteristic Functions, Chernoff Bounds, Conditional Expectations, Law of Large Numbers, Central Limit Theorem. Uniform, Binomial, Poisson and Normal Distributions.

Test for Large Samples: Testing of Hypothesis –Null and alternate hypothesis, level of significance and critical region-Z-test for single mean and difference of means, single proportion and difference of proportions.

Test for Small Samples: t-test for single mean and difference of means – F-test for comparison of variances, Chi-square test for goodness of fit, Chi-square test for independence.

Correlation and Regression: Correlation, lines of regression and examples.

Text books:

1.S. C. Gupta and V. K. Kapoor, Fundamentals of Mathematical Statistics, S. Chand and Co, 2006.

2. R. A. Johnson: Miller and Freund's Probability and Statistics for Engineers, Pearson Publishers, 9th Edition, 2017Thomas. G.B, and Finney R.L, Calculus, Pearson Education, 2007.

Reference books:

1.S. Milton and J. Arnold, Introduction to Probability and Statistics, Tata McGraw Hill Education Private Limited, 4th Edition, 2006.

2. R.K. Jain and S.R.K. Iyenger, Advanced Engineering Mathematics, 2nd Edition, Narosa Publishing House. 2005.

Ds152	**Linear Algebra** **3-0-3** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Solve linear equations using Gaussian elimination and echelon forms. 2. Understand matrix rank and use it to find solutions to linear equations. 3. Apply vector space concepts to solve problems in linear algebra. 4. Analyze linear transformations and their properties, such as rank-nullity theorem, matrix representation, and change of basis. 5. Understand inner product spaces, orthogonality, and eigen decomposition.
Course Outcomes	At the end of the course, students will be able to: 1. Solve a given system of linear equations 2. Determine a basis for a given vector space and find a direct sum of given subspaces 3. Find matrix representation of linear transformations and use linear transformations to model problems arising in engineering 4. Apply Gram Schimdt orthogonalization process for given set of vectors and obtain orthonormal set of vectors. 5. Apply spectral theorem for a given linear operator and determine whether a given linear operator/matrix is diagonalizable or not by using eigen values and eigen vectors.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2	3				
CO2	3	2		1			
CO3	3	1	2		1		
CO4	3			1	1		
CO5	3		3	2	2		

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Linear System of Equations: Gaussian Elimination, echelon forms, rank of a matrix, existence, uniqueness and multiplicity of solutions of linear equations.

Vector Spaces: Vector Space Definition, linear dependence and independence, spanning sets, basis, and dimension, definition of a subspace, intersection and sum of subspaces, direct sums.

Linear Transformations: Linear Transformation Definition, Range of Linear Transformation, Null Space of Linear Transformation, Rank – Nullity Theorem, Matrix representation of a linear transformation, change of basis, similarity transformation, invertible transformation, system of linear equations revisited, the four fundamental subspaces associated with a linear transformation.

Inner Product spaces: Inner Product Space Definition, induced norm, orthogonality, Gram-Schmidt orthogonalization process, orthogonal projections, unitary transformations and isometry. Eigen Decomposition: Eigenvalues and eigenvectors, characteristic polynomials and eigen spaces, diagonalizability conditions, invariant subspaces, spectral theorem.

Text books:

1. C. D. Meyer, "Matrix Analysis and Applied Linear Algebra," SIAM, 2000.
2. G. Strang, "Linear Algebra and its Applications," Cengage Learning, 4th Edition, 2005.

Reference books:

1. D. C. Lay, "Linear Algebra and its Applications," Pearson Education, 4th edition, 2011. .
2. V. Krishnamurthy, V. P. Mainra, J. L. Arora, An Introduction to Linear Algebra, Affiliated East- West Press Pvt Ltd, New Delhi, 1976.
3. Hoffman Kenneth and Kunze Ray, Linear Algebra, 2nd edition, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1971.

Others:

1. https://nptel.ac.in/courses/111106051

DS153	**Modern Physics**	**3-0-3**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To develop fundamental understanding of semiconductor properties. 2. To introduce the concept of charge carrier transport in semiconductors. 3. To understand the basic principles of p-n junctions and their applications. 4. To study the properties and applications of various semiconductor devices.		
Course Outcomes	By the end of the course, students should be able to: 1. Understand the basic concepts of semiconductor physics. 2. Analyze the behavior of charge carriers in semiconductors. 3. Design and analyze p-n junctions and related devices. 4. Identify the advantages and limitations of various semiconductor devices.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	2			1		
CO2	3	2			1		
CO3	3	2			1		

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1			3				
CO2			3				
CO3			3				

Detailed Syllabus

Introduction to Semiconductor Physics: Definition of Semiconductors, Types of Semiconductor, Energy Band Diagram, Carrier Concentration
Carrier Transport in Semiconductors: Diffusion and Drift, Mobility and Conductivity, Impurity Scattering
p-n Junctions: Formation of Junctions, Depletion Region, Current-Voltage Characteristics
Semiconductor Diodes: Ideal and Practical Diodes, Zener and Avalanche Breakdown, Diode Applications
Bipolar Junction Transistors: Structure and Operation, Current Gain, BJT Applications
Field Effect Transistors: Types of FETs, FET Characteristics, FET Applications, Optoelectronic Devices: LEDs, Photodiode, Solar Cells, Other Semiconductor Devices: Thyristors, Schottky Diodes, Varactors

Reference books:

1. Semiconductor Physics And Devices, Donald A. Neamen
2. Fundamentals of Semiconductor Devices, Betty Lise Anderson
3. Introduction to Semiconductor Devices: Diodes, Bipolar Junction Transistors, MOSFETs, and Optoelectronics, Kevin F. Brennan
4. Electronic Devices and Circuit Theory, Robert L. Boylestad and Louis Nashelsky
5. Semiconductor Optoelectronic Devices: Introduction to Physics and Simulation, Joachim Piprek

ME151	Concepts in Engineering Design 3-0-3 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To introduce to the undergraduate student the fundamental principles of Engineering Design 2. Learn about past design methods and product life cycle 3. Develop skills in needs and opportunities assessment and analysis 4. Learn different ideation techniques and methods 5. Develop skills in organizing design concepts and making design decisions.
Course Outcomes	1. The course should enable students to develop problem-solving skills in engineering design by exposing them to different approaches and methodologies used in the field. 2. The course should help students understand the application of engineering principles to real-world problems, and to design solutions that are both practical and efficient. 3. The course should introduce students to Computer-Aided Design (CAD) software and enable them to use it for engineering design tasks 4. The course should help students understand the importance of design constraints, and how to balance these constraints while designing a solution. 5. The course should enable students to communicate their design ideas effectively through technical reports, presentations, and visual aids.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	1	1	1				
CO2	2	2					
CO3	2		1				
CO4	2		1				
CO5	1	2	1				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Design Conceptualization and Philosophy, Original, Adaptive, Variant and Re-Design, Evolution of Concept, Need for Systematic design Past methods of and design Product life cycle, Innovation, Types of innovation.

Needs and opportunities, Vision and Mission of a concept, Type of needs, Technology S - curve, Need analysis, market analysis and competitive analysis, Kano Diagrams, SWOT analysis.

Conceptualization techniques – Idea generation – ideation, brainstorming, Trigger session Brain writing, Mind maps, SCAMPER, TRIZ, Biommicry, Shape mimicry, Familiarity Matrix Concepts screening, Concept testing - exploratory tests, Assessment tests, Validation tests Comparison tests – Case studies.

Organization of design concept and design methods, Engineering Design - Descriptive and prescriptive model, Design decisions and development of design Group work and case studies.

Text books:

1. Otto. K and Wood, K, Product Design, Pearson Education, 2001.

2. Pahl. G and Beitz. G, Engineering Design, Springer, 1996.

Reference books:

1. Ullman. D. G, The Mechanical Design Process, McGraw- Hill, 1997.

Others:

1.https://onlinecourses.nptel.ac.in/noc20_de10/preview

ME152	Design Realization Practice	0-3-2	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. Learn practical skills for machining product appearances with various tools and equipment. 2. Apply design principles and tools to realize successful product development. 3. Understand material properties and selection, as well as manufacturing processes. 4. Design for sustainability, ergonomics, and usability, and develop project planning, resource management, and team communication skills. 5. Study case examples and applications of design realization in various industries, while also gaining insight into cost estimation and quality control.		
Course Outcomes	1. Students will be able to apply their knowledge of design principles and tools to translate design ideas into physical prototypes and functional products. 2. Students will be able to understand the various stages involved in product design and development and their interdependence. 3. Students will gain an understanding of the various materials, their properties, and how to select the appropriate material for a given design requirement. 4. Students will develop skills in analyzing the functional requirements of a product and apply them in the design process to optimize the product's performance and usability. 5. Students will learn about different manufacturing processes and be able to select appropriate techniques to manufacture a product.		

Detailed Syllabus

The students are exposed to tools and equipments to machine external appearance of products of simple shapes. Wood carving, Plastic welding and cutting, engraving, sheet metal works, wire cutting are some of the process that the students will learn and use for product realization. The students will also be exposed high end machines to realize the product during demo sessions. Few sessions will be allocated to re-design an existing simple product in terms of shape, size functionality etc.

Overview of the product design and development process, Importance of design realization in product development, Design principles and tools

Properties of materials and their selection for a given design requirement, Manufacturing processes - casting, welding, machining, 3D printing, etc., Cost estimation and quality control

Design for assembly and disassembly Design for sustainability and environmental impact, Design for ergonomics and usability, Project planning and scheduling, Resource management - time, budget, materials, and personnel, Effective communication and collaboration in a team

Case studies of successful product design and realization, Applications of design realization in various fields - automotive, aerospace, consumer goods, etc.

Reference books:

1. Product Design and Development, Karl T. Ulrich and Steven D. Eppinger.
2. Engineering Cost Estimation and Control, Phillip F. Ostwald and Timothy S. McLaren
3. Project Management: A Managerial Approach, Jack R. Meredith and Samuel J. Mantel Jr.
4. Engineering Design: A Systematic Approach, Gerhard Pahl, Wolfgang Beitz, and Jörg Feldhusen.

Others:

1. https://onlinecourses.nptel.ac.in/noc20_me12/preview

DS154	Modern Physics Practice	0-3-2	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	1. Understand the particle nature of light and quantization of energy levels. 2. Impart knowledge about spectral series of hydrogen atom, speed of light and electron charge. 3. Gain knowledge about space quantization and electron spin 4. Learn the concepts of secure data transferring in quantum computing		

Detailed Syllabus

1. Black Body Radiation: Determination of Stefan's Constant Photoelectric Effect: Determination of Plank's constant.
2. Young's double-slit: Determination of wavelength of a laser source
3. Compton effect: Measure the energy of high-energy photons scattered from electrons in a brass rod as a function of the scattering angle.
4. Franck-Hertz Experiment: Demonstrate the quantization of atomic energy levels by bombarding the electrons and observing the difference in energy levels.
5. Hydrogen Spectra-Balmer Series Apparatus: determination of wavelengths of Balmer series from hydrogen emission spectra and find the Rydberg constant.
6. Millikan's oil drop experiment: Find the terminal velocity of the drop and the charge on a drop.
7. Foucault Method: Determination of the Speed of Light by the Foucault Method - Rotating Mirror Version
8. Davisson- Germen experiment: Verification of De-Broglie hypothesis and measure the wavelength of the electron.
9. Stern- Gerlach experiment: Verification of space quantization by measuring the angular momentum and magnetic moment of an atom.
10. Quantum Eraser: Demonstration of Quantum Eraser to confirm that the complementarity principle is indeed a fundamental part of quantum theory.
11. Quantum cryptography: Demonstrate the encryption of data.

Virtual experiments

12. Determination of Planck's constant using LED's of different wavelengths
13. Verification of Stefan's law
14. Study the emission spectra of Hydrogen and Neon vapours using a diffraction grating
15. Basics of Scanning Electron Microscopy: Secondary Electron
16. Understanding the quantum eraser concept and

Reference books:

1. Concepts of Modern Physics by Arthur Beiser, McGraw Hill Education, 2017.
2. Introduction to Quantum Mechanics by David J. Griffiths, McGraw Hill Education, 2017.
3. Berkeley Physics Course, Vol. 4: Quantum Physics, Eyvind H Wichman, McGraw Hill Education, 2017.
4. Quantum Computation and Quantum Information - Nielsen and Chuang, 10th Edition, Cambridge University Press, 2010.

Others:

1. https://www.vlab.co.in/
2. https://lab.quantumflytrap.com

XXXX	Digital Logic Design	2-3-4	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To teach various number systems, binary codes and their applications 2. To introduce K-MAPs to simplify Boolean expressions 3. To gain an understanding of combinational and sequential logic circuits and the differences between them 4. To make students familiar with the different types of digital logic families, their characteristics, and their applications		
Course Outcomes	At the end of the course students will be able to: 1. Define various number systems, including 2's complement representation and perform arithmetic operations using this representation 2. Use Karnaugh maps for simplifications 3. Understand the design procedures of combinational and sequential circuits 4. Identify the importance of canonical forms in the minimization and other optimization of Boolean formulas in general and digital circuits 5. Analyze and design digital systems using various logic families, including TTL and CMOS		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	1	2				
CO2	2	2	2				
CO3	2	1	3	1			
CO4	2	2	1				
CO5	1	1	3	2			

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Digital Systems: Binary Numbers, Number-Base Conversions, Octal and Hexadecimal Numbers, Signed Binary Numbers-1's and 2's complement, Binary Codes, BCD, Gray, Excess 3, Binary Storage and Registers. Boolean Algebra and Logic Gates: Basic Definitions, Basic Theorems and Properties, Boolean Functions, Canonical and Standard Forms, Other logic operations, Introduction of HDLs and their syntax.

Simplification of Boolean expressions – Algebraic methods and canonical forms, Gate level minimization: The K-map method, three-variable map, four-variable map, five-variable map, product of sums simplification, don't-care conditions, NAND and NOR implementation. Quine-McCluskey algorithm, Determination and selection of Prime Implicants, Essential and Non essential prime Implicants.

Combinational Logic, Implementing Combinational Logic, Binary Adder- Subtractor, Decimal Adder, Binary Multiplier, Parallel adders and look-ahead adders, Comparator, Decoders, Encoders, Multiplexers, Demultiplexers, Parity generators and checkers.

Sequential Circuits: Latches and Flip-Flops: SR, JK, D, T; Excitation tables, Shift Registers, Design of counters, Ripple Counters, Synchronous Counters, Memory Units.

Integrated Circuits: MOS, CMOS, RTL, DTL, TTL, ECL, Programmable Logic Devices, Designing Combinational Logic Circuits with PLDs, Random Access Memory (RAM), Read-Only Memory (ROM), Programmable Logic Arrays (PLA), Programmable Array Logic (PAL), Content Addressable Memory.

Text books:

1. Mano, M. Morris. Digital logic and computer design. Pearson Education India, 2017.

Reference books:

1. Floyd, Thomas L. Digital fundamentals, 10/e. Pearson Education India, 2011.
2. Wakerly, John F. Digital Design: Principles and Practices, 4/E. Pearson Education India, 2008.
3. Kime, Charles R., and M. Morris Mano. Logic and computer design fundamentals. Prentice Hall, 2003.
4. Widmer, Neal S., and Ronald J. Tocci. Digital systems: Principles and applications. Prentice Hall. Pearson Education International, 2007.
5. Roth Jr, Charles H., Larry L. Kinney, and Eugene B. John. Fundamentals of logic design. Cengage Learning, 2020.

Others:

1. https://nptel.ac.in/courses/117105080

CS152	**Python Programming Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To know the basics of algorithmic problem-solving. 2. To read and write simple Python programs. 3. To develop Python programs with conditionals and loops. 4. To define Python functions and call them. To use Python data structures – lists, tuples, dictionaries. 5. To do input/output with files in Python.		
Course Outcomes	1. Develop algorithmic solutions to simple computational problems. 2. Read, write, execute by hand simple Python programs. 3. Structure simple Python programs for solving problems. 4. Decompose a Python program into functions and Represent compound data using Python lists, tuples, and dictionaries. 5. Read and write data from/to files in Python Programs.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2	3	2								2	2	2
CO2	1	3	2	3								2	1	2
CO3	1	1	3	2								1	3	2
CO4	2	1		1	1								1	
CO5			3	3	3								2	1

Detailed Syllabus

Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.
Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.
Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing – list comprehension; Illustrative programs: selection sort, insertion sort, mergesort, histogram. Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file.

Reference books:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016
2. Guido van Rossum and Fred L. Drake Jr, "An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.
3. Charles Dierbach, "Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
4. John V Guttag, "Introduction to Computation and Programming Using Python", Revised and expanded Edition, MIT Press, 2013.
5. Kenneth A. Lambert, "Fundamentals of Python: First Programs", CENGAGE Learning, 2012.
6. Paul Gries, Jennifer Campbell and Jason Montojo, "Practical Programming: An Introduction to Computer Science using Python 3", Second edition, Pragmatic Programmers, LLC, 2013.
7. Robert Sedgewick, Kevin Wayne, Robert Dondero, "Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
8. Timothy A. Budd, "Exploring Python", Mc-Graw Hill Education (India) Private Ltd.,, 2015

XXXX	**NSS/NCC/NSO** **0-0-0** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To shape the students into strong, disciplined, responsible, and highly motivated citizens 2. Foster in young minds a sense of dedication to the values of national integration, secularism, nationalism, and social welfare 3. To educate the students about physical fitness 4. To emphasises working democratically and uphold the values of unselfish service, respect for another person's viewpoint, and care for fellow humans
Course Outcomes	1. Students develop empathy, and contribute to social causes, leading to a better understanding of societal needs and issues. 2. Students develop leadership skills, communication skills, teamwork, and critical thinking abilities, which are essential for personal growth and career development 3. Students learn about their social responsibility towards nation, and develop a sense of civic duty. 4. Students will be able identify social issues and take proactive measure towards finding solutions. 5. Students develop a sense of responsibility towards the well-being of the community.

Activities:

National Cadet Corps (NCC) conduct atleast 20 parades each year. Conduct social welfare awareness, first aid, and personal hygiene programmes
Conduct sports activities and participate in inter-institutional sports activities
Camps are organised in neighbouring villages where efforts are made to better the people' circumstances. A volunteer with the NSS fills their calendar with events like cleaning up temples, running blood donation camps, and teaching school children how to use computers.

Semester III

DS201	**Random Variables and Stochastic Processes (to be updated)**	**3-0-3**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To understand the fundamental concepts of random variables and their distributions. 2. To analyze and characterize stochastic processes and their behavior over time. 3. To apply theoretical knowledge to solve real-world problems involving uncertainty and randomness. 4. To develop skills in modeling and analyzing systems using stochastic methods. 5. To familiarize with advanced topics in stochastic processes and their applications in various fields.		
Course Outcomes	1. Students will be able to define and explain the basic concepts of random variables, probability distributions, and their properties. 2. Students will develop the ability to analyze and characterize various types of stochastic processes. 3. Students will be able to apply stochastic methods to model and solve practical problems involving randomness. 4. Students will gain experience in using mathematical software and tools to simulate and analyze stochastic systems. 5. Students will explore and understand advanced topics and applications of stochastic processes in diverse areas such as finance, engineering, and natural sciences.		

Detailed Syllabus

Introduction to Probability, Random Variables, Probability Distribution Functions, Cumulative Distribution Functions, Joint Distributions, Conditional Distributions, Expectation and Variance, Moment Generating Functions, Inequalities and Limit Theorems, Discrete Stochastic Processes, Continuous Stochastic Processes, Poisson Process, Renewal Theory, Markov Chains, Classification of States in Markov Chains, Steady-State Behavior, Birth-Death Processes, Queueing Theory, Martingales, Brownian Motion, Stochastic Differential Equations, Applications to Finance, Applications to Engineering, Applications to Biology, Time Series Analysis, Stationarity and Ergodicity, Autocorrelation and Power Spectral Density, Wiener Process, Stochastic Integration, Filtering and Prediction, Kalman Filter, Monte Carlo Methods.

Reference books:

1. R. S. Ross, Introduction to Probability and Statistics for Engineers and Scientists, Academic Press, 2014
2. R. A. Johnson and C. B. Gupta, Miller Freund's Probability and Statistics for Engineers, 7/e, Pearson Education Inc, 2005.
3. "Stochastic Processes" by J. Medhi
4. "Essentials of Stochastic Processes" by Richard Durrett
5. "Adventures in Stochastic Processes" by Sidney I. Resnick

CS201	**Discrete Mathematics**	**3-0-3**	**Core**
Pre-requisite	Basic Mathematical and Rational Thinking		Effective from: 12/4/2023
Course Objectives	1. To introduce concepts of mathematical logic for analyzing propositions and proving theorems. 2. To represent facts/events using first order logic and apply rules of inferences to draw conclusions. 3. To understand basic mathematical structures such as sets, relations and functions and their properties. 4. To apply basic counting techniques to solve combinatorial problems. 5. To analyze a real-world problem, represent it as a graph theoretic problem and find a solution to that problem using graph theoretic principles.		
Course Outcomes	At the end of the course, student will be able to: 1. Analyze logical propositions via truth tables. 2. Prove mathematical theorems using different techniques. 3. Work on sets, determine properties of relations, and identify equivalence and partial order relations. 4. Evaluate combinations and permutations on sets. 5. Understand the real-time usage of graphs and use it for solving real-life problems.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2	2	2	2	1		
CO2	1	2	2	3			
CO3	2	2	1				
CO4	2	2	1	1			
CO5	3	3	2	2		2	

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						1	1
CO2						2	
CO3						1	2
CO4						1	
CO5						2	2

Detailed Syllabus

The Basics of Logics and Proofs: Propositions, predicates, quantifiers, first order logic, rules of inference, introduction to proofs, methods of proofs.

Set Theory and Relations: Sets and operations, relations, different relations, partial orderings, functions, sequences and summations, growth of functions.

Induction, Recursion and Counting: Mathematical induction, recursion, Pigeonhole principle, permutations and combinations.

Probability and Boolean Algebra: Discrete probability, Bayes' theorem, expected value and variance, boolean functions, minimization of circuits.

Graphs and Trees: Basics of graph theory, connectivity, Euler and Hamilton paths, shortest path problems, planarity, coloring, tree traversals, spanning Trees.

Text books:

1. K. H. Rosen, "Discrete Mathematics and its Applications," McGraw Hill, 6th Edition, 2007.

Reference books:

1. D. F. Stanat and D. F. McAllister, "Discrete Mathematics in Computer Science," Prentice Hall, 1977.

2. R. L. Graham, D. E. Knuth, and O. Patashnik, "Concrete Mathematics," Addison Wesley, 1994.

3. Busby, Kolman, and Ross, "Discrete Mathematical Structures," Pearson Education India, 6th Edition, 2015.

4. C. L. Liu, "Elements of Discrete Mathematics," Tata McGraw Hill, 1995.

CS202	**Data Structures** **2-2-4** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. To familiarise the concepts of simple and abstract data representations in computer systems. 2. To introduce the different concepts of data manipulation, searching, sorting and retrieval. 3. To manifest the real-life problem solving through graphs and trees. 4. To develop intermediate programming skills required for handling data operations in computers. 5. To develop basic analysis skills for evaluating various techniques to solve the same problem (like sorting, searching, etc.

CO-PO/PSO Mapping

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	**3**					1		1			1			**2**
CO2	**1**		**2**	**1**	**2**					2	1		**3**	**2**
CO3	1	**3**	**2**	**3**								2	**2**	**2**
CO4	**2**	2	**3**	**2**	2								**2**	**3**
CO5	2	**1**	2	**3**		1					1		2	**3**

Detailed Syllabus

Introduction to data structures: basic introduction, categories of data structures, data representation with structures and unions, abstract data type (ADT), implementation of ADTs using array, introduction to linked list, doubly linked lists, circular linked list, implementation of ADTs using linked lists, conceptual understanding of recursive function using stacks, principles of stacks (LIFO) and queues (FIFO), balancing of parenthesis, operator precedence, evaluation and interconversion of infix, prefix and postfix expression.

Searching and sorting techniques: linear Search, binary Search, sorting: internal vs external, stable sorting, non-comparison sorting: bucket sort, radix sort, comparison sorting: insertion sort, selection sort, bubble sort, merge sort, quick sort, etc.

Tree data structure: nodes and their properties, height and depth of a tree, representation of tree data structure, ordered trees, tree traversals, tree operations: insertion, deletion, searching, binary trees, threaded binary trees, m-ary trees, binary search trees, AVL trees, red-black tree, B-trees, B+trees, decision trees, heaps, forests and Huffman coding.

Graphs: Introduction to graphs, representation of graphs, terminologies: node, edge, path, walk, cycle, circuit, directed vs undirected graphs, complete graph, bipartite graph, cliques, wheels, pendant, cut set, etc., planar graph, isomorphic graphs, introduction to graph colouring problem and their applications, Dirac's and Ore's theorem.

Applications of graphs: shortest path problems and their importance in real life, Dijkstra's algorithm, traveling salesman problem, Kuratowski's theorem, depth first and breadth first searches, spanning trees and their importance, algorithms for finding the minimum spanning tree from graphs.

Text books:

1. Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. "Introduction to algorithms". MIT press, 2022.
2. Lipschutz, S., "Data Structures". Schaum's Outline Series, McGraw Hills, 2022.
3. Rosen, Kenneth H. Discrete mathematics applications. McGraw-Hill, 1999.

Reference books:

1.Kruse, R. and Tondo, C.L. "Data structures and program design in C". Pearson Education India, 2007.

Others:

1.https://nptel.ac.in/courses/106102064

CS203	**Computer Organization and Architecture**	**3-0-3**	**Core**
Pre-requisite	Digital Logic Design		Effective from: 12/4/2023
Course Objectives	1. To introduce various aspects of computer organization such as instruction format, addressing modes, etc. 2. To understand the functions and interactions of the major hardware components of a computer system 3. To understand the impact of computer organization on system performance and design 4. To understand the role of memory and storage in computer systems 5. To understand the basics of input/output operations and their interfaces with the computer system		
Course Outcomes	Students will be able to: Understand the key components of a basic computer, CPU and execution of instructions 2. Analyze the execution and time taken by instructions in a pipelined processor and visualization with modern tools like RIPES 3. Understand the functionality of memory, I/O devices and importance of memory hierarchy, cache. 4. Design the effective CPU in terms of speed, technology, cost, performance 5. Assemble language Coding to solve the complex problem with RISC, CISC instruction sets		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3						
CO2		3			2		
CO3	3						
CO4			3				
CO5				3			-
CO	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1							
CO2						3	3
CO3							3
CO4						3	3
CO5						3	

Detailed Syllabus

Role of abstraction, basic functional units of a computer, Von-Neumann model of computation, A note on Moore's law, Notion of IPC, and performance. Data representation and basic operations.
Instruction Set Architecture (RISC-V):
CPU registers, instruction format and encoding, addressing modes, instruction set, instruction types, instruction decoding and execution, basic instruction cycle, Reduced Instruction Set Computer (RISC), Complex Instruction Set Computer (CISC), RISC-V instructions; X86 Instruction set.
The Processor:
Revisiting clocking methodology, Amdahl's law, Building a data path and control, single cycle processor, multi-cycle processor, instruction pipelining, Notion of ILP, data and control hazards and their mitigations
Memory hierarchy:
SRAM/DRAM, locality of reference, Caching: different indexing mechanisms, Trade-offs related to block size, associativity, and cache size, Processor-cache interactions for a read/write request, basic optimizations like writethrough/write-back caches, Average memory access time, Cache replacement policies (LRU), Memory interleaving. Storage and I/O:
Introduction to magnetic disks (notion of tracks, sectors), flash memory. I/O mapped, and memory mapped I/O. I/O data transfer techniques: programmed I/O, Interrupt-driven I/O, and DMA.
Superscalar processors and multicore systems:
Limits of ILP, SMT processors, Introduction to multicore systems and cache coherence issues

Text books/References:

1. "Computer Organization and Design: The Hardware/Software Interface", David A. Patterson and John L. Hennessy, 5th Edition, Elsevier.
2. "Computer Organisation and; Architecture", Smruti Ranjan Sarangi, McGraw Hill
3. "Computer System Architecture", Mano M. Morris, Pearson.
4. "Computer Organization and Embedded Systems", 6th Edition by Carl Hamacher, McGraHill Higher Education
5. "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.
6. "Computer System Design and Architecture", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

Others:

1. http://web.cecs.pdx.edu/ harry/riscv/RISCV-Summary.pdf

CS204	Design and Analysis of Algorithms	3-0-3	Core
Pre-requisite	Data Structures		Effective from: 12/4/2023
Course Outcomes	CO1: Develop the skill of understanding a given problem and design an algorithm for the same. CO2: Apply various algorithm design techniques to solve computational problems efficiently with the time and space complexity of algorithms using asymptotic notations. CO3: Students should develop basic knowledge of a wide range of algorithm design techniques including divide and conquer, greedy, dynamic programming, branch and bound, and apply them to problems and find solutions. Analyse. CO4: Design and implement graph algorithms for problems such as shortest path and minimum spanning tree. CO5: Evaluate the correctness and efficiency of algorithms using formal analysis techniques and Understand the notion of solvable, NP, NP-hard, and NP-complete problems.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	3	2	2	2	1	
CO2	3	3	3	2	3	2	
CO3	3	3	3	3	3	2	
CO4	3	3	3	3	3	2	
CO5	3	3	3	3	3	2	
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1		1	2	1	2	3	3
CO2		2	3	2	2	3	3
CO3		2	3	2	3	3	3
CO4		2	3	2	2	3	3
CO5		2	3	2	3	3	3

Detailed Syllabus

Introduction to Algorithms and Complexity: Overview of Algorithms: Definition and Importance Analysis of Algorithms: Time Complexity, Space Complexity Asymptotic Notations: Big O, Big Omega, Big Theta. Solving Recurrence Relations: Substitution Method, Recurrence Tree Method, Master Theorem Incremental and Decremental Algorithm Design Strategies Case Studies and Lower Bound for Sorting.

Algorithm Design Techniques: Divide and Conquer: Merge Sort, Quick Sort, Median Finding, Greedy Algorithms: Fractional Knapsack Problem, 0/1 Knapsack Problem, Minimum Spanning Trees: Prim's Algorithm, Kruskal's Algorithm, Huffman Coding, Sets of intervals.

Dynamic Programming and Graph Algorithms: Dynamic Programming: Longest Common Subsequence (LCS), Matrix Chain Multiplication, 0/1 Knapsack Problem. Graph Algorithms: Graph Representation-Adjacency Matrix, Adjacency List, Topological Sorting Shortest Path Algorithms: Dijkstra's Algorithm, Bellman-Ford Algorithm, Floyd-Warshall Algorithm for All-Pairs Shortest Paths.

Advanced Topics in Algorithms: Backtracking: N-Queens Problem, Subset Sum Problem Branch and Bound: Job Scheduling Problem Complexity Theory: Introduction to NP-Completeness, Complexity Classes: P, NP, NP-Hard, NP- Complete, Reductions and Satisfiability Problems, Examples of NP-Complete Problems: Satisfiability Problem, Independent Set, Vertex Cover, Hamiltonian Circuit

Reference books:

1. E. Horowitz, S. Sahni, and S. Rajasekaran, "Computer Algorithms," 2nd Edition, Galgotia Publications, 2007.

2. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms," Prentice Hall India, 2nd Edition, 2001.

3. Aho, Hopcroft, and Ullmann, "Data Structures amp; Algorithms," Addison Wesley, 1983.

4. C. H. Papadimitriou, Computational Complexity, Addison-Wesley Publishing Company, 1994.

5. D. S. Garey and G. Johnson, Computers and Intractability: A Guide to the Theory of NP- Completeness, Freeman, New York, 1979.

CS205	Data Structures Practice	0-3-2	Core
Pre-requisite	Basic Programming Knowledge	Effective from: 12/4/2023	
Course Outcomes	1. Understand the use of basic programming paradigms like structures and pointers in implementing abstract data types 2. Implement different linear and non-linear data structures with a thorough understanding of their basic operations like insertion or deletion in specific positions or segments 3. Develop coding skills for the implementation of various sorting and searching algorithms and reflect on their usability under different constraints 4. Analyze the necessity of height-balanced non-linear data structures based on their running time, space requirements, and ease of operations 5. Ideate about the data structures by comprehending their suitability for different purposes like searching, sorting, hashing		

CO-Po/PSO mapping

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2
CO1	**2**		1		2								**2**	**3**
CO2	**3**	**2**	**3**	**2**									**2**	**2**
CO3		**3**	**2**	**3**									**3**	**3**
CO4	**1**	2		**1**								1	**2**	**3**
CO5			3	**3**	1							**2**	3	**3**

List of Experiments

Syllabus/ List of experiments

Implementation of linear data structures like arrays and performing basic operations like insertions, deletions, concatenation, etc.

Implementation of different comparison-based sorting algorithms like bubble sort, insertion sort, selection sort

Contrast between linear versus binary search in terms of implementation Implementation of linked lists: highlighting their advantages and disadvantages when compared to arrays

Operations in linked lists such as insertion, deletion, concatenation and implementation of singly linked list, doubly linked list and circular linked list

Implementation of non-linear data structures: trees and operations like insertion, deletion, rearrangements

Implementation of tree traversal algorithms like level-order, pre-order, post-order, etc.

Implementation of complex sorting algorithms like quick sort, merge sort, tree sort

Implementation of binary search trees with their operations

Implementation of graphs through adjacency list and adjacency metrics with their traversals

Implementation of hashing with collision resolution techniques

Reference books:

1."Data Structures and Algorithms in C++" by Adam Drozdek

2. "Data Structures and Algorithms Made Easy: Data Structure and Algorithmic Puzzles" by Narasimha Karumanchi

3."Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein

4."Data Structures Using C" by Reema Thareja

5. "Data Structures and Algorithms in Java" by Robert Lafore.

CS206	**Computer Organization and Architecture Practice**	**0-3-2**	**Core**
Pre-requisite	Digital Logic Design		Effective from: 12/4/2023
Course Objectives	1. Understanding architecture and instructions through assembly programming 2. Understanding performance issues related to pipelining and cache using architectural simulator 3. Understanding memory access patterns and changing basic cache memory parameters to analyze the impact of standard programs or benchmarks using architectural simulators.		
Course Outcomes	Students will gain: 1. Expertise in assembly language programming on simulator 2. Ability to debug complex code that interacts with hardware at a low level 3. Understand the execution and time taken by instructions in a pipelined processor and visualization with modern tools like RIPES 4. Analyze the cache memory performance with various configurations		

List of Experiments

Write programs in ARM/RISC V assembly language and test these on an instruction set simulator. Typical examples are given below. Some of these are dependent on I/O facilities provided by the simulator.
Generate some interesting numbers (example - Happy numbers, Autonomic numbers, HardyRamanujan numbers etc.)
Implement a 4-function calculator
Sort an integer array using merge sort (recursive)
Evaluate an arithmetic expression specified as a string (using recursive functions)
Implement a simple game

Usage of an instruction pipeline visualization tool like RIPES
Write or generate sequence of instructions and observe the overall pipeline stalls
with and without data hazards, control hazards, and with/without data forwarding.
Rearrange the sequence of instructions or the program so that the pipeline stalls will be minimized.

Configure the simulator [gem5 is preferred] to operate on the binaries of the benchmark as the input.
Run the program and examine the IPC, cache hit rate, number of conflict misses and block replacements.
Vary the cache size, block size, and associativity and analyze the metrics and reason the changes observed.

Modify the block replacement algorithms and see the impact at cache memory performance
Calculate the access time, power and are associated with a given cache configuration.
Vary the cache size, block size, and associativity and analyze the metrics and reason the changes observed.

Text books/References:

1. “Computer Organization and Design: The Hardware/Software Interface”, David A. Patterson and John L. Hennessy, 5th Edition, Elsevier.
2. “Computer Organisation and; Architecture”, Smruti Ranjan Sarangi, McGraw Hill
3. “Computer System Architecture”, Mano M. Morris, Pearson.
4. “Computer Organization and Embedded Systems”, 6th Edition by Carl Hamacher, McGraHill Higher Education
5. “Computer Architecture and Organization”, 3rd Edition by John P. Hayes, WCB/McGraw-Hill “Computer Organization and Architecture: Designing for Performance”, 10th Edition by William Stallings, Pearson Education.
6. “Computer System Design and Architecture”, 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

Others:

1. http://web.cecs.pdx.edu/ harry/riscv/RISCV-Summary.pdf

CS207	**Design and Analysis of Algorithms Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	CO1: Develop the skill of understanding a given problem and design an algorithm for the same. CO2: Apply various algorithm design techniques to solve computational problems efficiently with the time and space complexity of algorithms using asymptotic notations. CO3: Students should develop basic knowledge of a wide range of algorithm design techniques including divide and conquer, greedy, dynamic programming, branch and bound, and apply them to problems and find solutions. Analyse. CO4: Design and implement graph algorithms for problems such as shortest path and minimum spanning tree. CO5: Evaluate the correctness and efficiency of algorithms using formal analysis techniques and Understand the notion of solvable, NP, NP-hard, and NP-complete problems.		

List of Experiments

1. Implement the algorithms to perform the Bubble sort, Insertion sort for any given list of numbers.
2. Apply Greedy method to compress the given data using Huffman encoding.
3. Implement fractional knapsack problem using Greedy Strategy.
4. Implement the Divide and Conquer methods – Merge Sort, Quick Sort.
5. Implement minimum spanning tree using Prim's algorithm and analyse its time complexity.
6. Apply dynamic programming methodology to implement 0/1 Knapsack problem.
7. Solve the longest common subsequence problem using dynamic programming.
8. Find the length of the longest subsequence in a given array of integers such that all elements of the subsequence are sorted in strictly ascending order.
9. Apply dynamic programming methodology to find all pairs shortest path of a directed graph using Floyd's algorithm.
10. Linear time DFS and BFS implementation with adjacency list representation.
11. Kruskal's algorithm implementation in O((n+e)log n) complexity.
12. Prim's algorithm implementation in O((n+e)log n) complexity.
13. Dijkstra's algorithm implementation in O((n+e)log n) complexity.
14. Write a program to solve N-QUEENS problem.
15. Write a program to solve Sum of subsets problem for a given set of distinct numbers.

Reference books:

1. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, Introduction to Algorithms, PHI, 1998.
2. S. Sahni, Data Structures, Algorithms, and Applications in C++, McGraw Hill, 1998

Semester IV

CS251	**Automata Theory and Compiler Design** **3-0-3** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	At the end of the course, students will be able to: 1. Understand the fundamental concepts of automata theory and formal languages. 2. Apply the principles of automata theory to solve computational problems. 3. Design lexical analysers using regular expressions and finite automata. 4. Construct syntax analysers using context-free grammars and parsing algorithms. 5. Perform semantic analysis and generate intermediate code for a given source program. 6. Optimize intermediate code and generate machine code for target architectures.

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2	2	2	1	2		
CO2	3	3	2	1	3		
CO3	3	3	3	1	3		
CO4	3	3	3	2	3		
CO5	3	3	3	2	3		
CO6	3	3	3	2	3		
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	2	1	1	2	2
CO2	1	2	3	2	1	3	3
CO3	1	2	3	2	1	3	3
CO4	1	2	3	2	1	3	3
CO5	1	2	3	2	1	3	3
CO6	1	2	3	2	1	3	3

Detailed Syllabus

Module 1: Introduction to Automata Theory, Finite Automata and Regular Languages Introduction: Notion of formal language- Strings, Alphabet,Language, Operations, Finite State Machine, definitions, finite automaton model, acceptance of strings and languages, deterministic finite automaton, equivalence between NFA and DFA, Conversion of NFA into DFA, minimization of FSM, equivalence between two FSMs, Moore and Mealy machines. Regular expressions: Regular sets, regular expressions, identity rules, manipulation of regular expressions, equivalence between RE and FA, inter conversion, Pumping lemma, Closure properties of regular sets regular grammars, right linear and left linear grammars equivalence between regular linear grammar and FA, inter conversion between RE and RG.

Module 2: Context-Free Grammars and Languages, Turing Machines and Decidability Context free grammars: Derivation, parse trees. Language generated by a CFG. Eliminating useless symbols, productions, and unit productions. Chomsky Normal Form. Pushdown automata: Definition, instantaneous description as a snapshot of PDA computation, notion of acceptance for PDAs. Pumping lemma for Context free grammars, Turing machine: Turing machine, definition, model, design of TM, Computable Functions, recursive enumerable language, Church's Hypothesis, Counter machine, types of TMs.

Module 3: Introduction to Compiler Design, Lexical Analysis, Syntax Analysis Overview of Compilation Process, Structure of a Compiler, Phases of Compilation, Role of Lexical Analyzer, Tokenization and Lexemes, Regular Expressions and Finite Automata, Construction of Lexical Analyzers, Lexical Analyzer Tools (e.g., Lex), Role of Syntax Analyzer, Context-Free Grammars (CFG), Ambiguity in Grammars, Parsing Techniques: Top-Down Parsing, Bottom-Up Parsing, LL arsing, LR Parsing, Syntax Analyzer Tools (e.g., Yaac).

Module 4: Semantic Analysis, Intermediate Code Generation Role of Semantic Analyzer, Syntax-Directed Definitions and Translation, Type Checking and Type Systems, Intermediate Code Generation: Intermediate Representations (IR), Syntax-Directed Translation Schemes Three-Address Code Generation, Quadruples and Triples, Translation of Expressions and Control Structures Intermediate Code Generation Techniques

Module 5: Code Optimization, Code Generation Principles of Code Optimization, Optimization Techniques: Constant Folding, Strength Reduction, Loop Optimization, Data Flow Analysis, Code Generation: Role of Code Generator, Code Generation Techniques: DAG Representation, Peephole Optimization, Register Allocation Techniques, Machine-Independent and Machine-Dependent Code Generation

Reference books:

1. J Hopcroft, JD Ullman and R Motwani, Introduction to Automata Theory, Languages and Computation, Third Edition, Pearson, 2008.

2. Alfred V. Aho, Jeffrey D Ullman, S. Lam, and Ravi Sethi, "Compilers: Principles, Techniques and Tools", Pearson Education, 2015

3. M Sipser, Introduction to the Theory of Computation, Second Edition, Thomson, 2005.

4. Lewis H.P. and Papadimition C.H. Elements of Theory of Computation, Prentice Hall of India, Fourth Edition, 2007. 4. S. Arora and B. Barak, Computational Complexity: A Modern Approach, Cambridge University Press, 2009.

5. C. H. Papadimitriou, Computational Complexity, Addison Wesley Publishing Company, 1994.

6. D. C. Kozen, Theory of Computation, Springer, 2006. 7. D. S. Garey and G. Johnson, Computers and Intractability: A Guide to the Theory of NP- Completeness, Freeman, New York, 1979.

7. Alfred Aho, Ravi Sethi and Jeffrey D Ullman, Compilers Principles, Techniques and Tools, Pearson Education, 2003.

8. Levine J.R, Mason T, Brown D, Lex amp; Yacc, O'Reilly Associates, 1992.

9. Allen I. Holub, Compiler Design in C, Prentice Hall, 2003.

10. Kamala Krithivasan and R Rama, Introduction to Formal Languages, Automata Theory and Computation, Pearson Education, 2009.

CS252	**Artificial Intelligence 3-0-3** **Core**
Pre-requisite	Algorithms, Problem solving Effective from: 12/4/2023
Course Outcomes	1. Learn various agent models. 2. Learn and implement various searching techniques on live examples. 3. Create and design knowledge-based agents. 4.Create and design agents to work in uncertain environments. 5. Perform statistical learning on real world problems.

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2	1	1	1	1		
CO2	3	3	2	2	1		
CO3	1	2	1	2	2		
CO4	1	2	1	2	1		
CO5	2	1	1	1	1		
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						1	1
CO2						2	1
CO3						2	2
CO4						3	3
CO5						2	2

Detailed Syllabus

Introduction – Agents and Objects – Evaluation of Agents – Agent Design Philosophies - Multi- agent System – Mobile Agents – Agent Communication – Knowledge query and Manipulation Language – Case Study. What is AI? , The Foundations of Artificial Intelligence

Intelligent Agents – Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents

Solving Problems By Search – Problem-Solving Agents, Formulating problems, Searching for Solutions, Uninformed Search Strategies, Breadth-first search, Depth-first search, Searching with Partial Information, Informed (Heuristic) Search Strategies, Greedy best-first search, A* Search: Minimizing the total estimated solution cost, Heuristic Functions, Local Search Algorithms and Optimization Problems, Online Search Agents and Unknown Environments

Adversarial Search – Games, The minimax algorithm, Optimal decisions in multiplayer games, Alpha-Beta Pruning, Evaluation functions, Cutting off search, Games that Include an Element of Chance

Logical Agents – Knowledge-Based agents, The Wumpus World, Logic, Propositional Logic: A Very Simple Logic, Reasoning Patterns in Propositional Logic, Resolution, Forward and Backward chaining

First Order Logic – Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic Inference In First Order Logic – Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution

Uncertainty – Acting under Uncertainty, Basic Probability Notation, The Axioms of Probability, Inference Using Full Joint Distributions, Independence, Bayes' Rule and its Use, The Wumpus World Revisited

Probabilistic Reasoning – Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distribution, Exact Inference in Bayesian Networks, Approximate Inference in Bayesian Networks

Statistical Learning Methods – Statistical Learning, Learning with Complete Data, Learning with Hidden Variables: EM Algorithm.

Text books:

1. Russel,S., and Norvig,P., (2015), Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall
2. Nils J Nilsson, Artificial Intelligence: A New Synthesis, Morgan Kaufmann Publications, 2000.

Reference books:

1. Kevin Knight, Elaine Rich and Shivshankar B. Nair, Artificial Intelligence, McGraw Hill, 2017, 3rd Edition.

CS253	Operating Systems 3-0-3 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To understand the purpose, structure and functions of operating systems. 2. To understand what a process is and how processes are synchronized and scheduled. 3. To understand different approaches to memory management. 4. To understand the structure and organization of the file system. 5. To know how to design a simple operating system.
Course Outcomes	At the end of the course, student will be able to: 1. Understand the architecture and functionalities of modern OS. 2. Identify the problems related to process management and synchronization and apply learned methods to solve basic problems. 3. Apply semaphores and monitors for classical and real-world synchronization scenarios. 4. Comprehend the cause and effect related to deadlocks and to analyse them related to common circumstances in operating systems. 5. Perceive the use of virtual memory in modern operating systems and the structure of the most common file-systems.

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	1	1	1								1	1
CO2	3	3	2	2	1								2	1
CO3	1	2	1	2	2								2	2
CO4	1	2	1	2	1								3	3
CO5	2	1	1	1	1								2	2

Detailed Syllabus

Introduction: Evolution of operating system, Batch, multi-programming, time sharing, multiprocessor, distributed, cluster and real-time systems, Linux system introduction and commands, Operating system structures: Computer system structure, Network structure, I/O Structure, Storage Structure, System components, Operating-System Services, System Calls, Types of System Calls, System Programs, System structure, Virtual Machines, System Design and Implementation.

Process Management: Process Concept, Process Control Block and attributes. Process States and Multi-programming, Process state transition diagram and various process schedulers, Threads, CPU scheduling algorithms: Scheduling Criteria, First Come First Serve, Shortest Job First, Shortest Remaining Time First, Round Robing scheduling algorithm, Longest Job First, Longest Remaining Time First, Highest Response Ration Next, Priority scheduling (Preemptive and Non preemptive), Multilevel queue scheduling, Multilevel feedback queue scheduling.

Process Synchronization and Deadlocks: Synchronization Background, Race condition, Critical section problem: Mutual Exclusion, Progress, Bounded Waiting. Peterson solution problem. Semaphores introduction: Wait, Signal, Binary semaphores. Semaphore and Monitors: Producer consumer problem, Reader writer problem, Dining philosopher's problem. Deadlock: Introduction to Deadlock, Deadlocks in resource allocation graphs: Single instance and Multiple instance, Necessary conditions for Deadlock, Deadlock handling methods, Deadlock Prevention, Deadlock Avoidance: Bankers algorithm, Deadlock Detection and Recovery.

Memory Management: Background, Swapping, Logical Address Space and Physical Address Space, Contiguous Memory Allocation:First Fit, Best Fit, Worst Fit, Internal and External Fragmentation, Non-Contiguous Memory Allocation: Paging, Structure of the Page Table, Multilevel Paging, Inverted Paging, Segmentation, Segmentation with Paging, Address Translations, Virtual Memory, Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing, Memory-mapped I/O.

File and Disk Management: File Concept, File Attributes, File Access Methods, Directory Structure, File System Organization and Mounting, File Sharing, Protection File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance. Storage Management: Disk Structure, Logical and Physical View, Disk Scheduling Algorithms: First Come First Serve, Shortest Seek Time First, SCAN, C-SACN, LOOK, C-LOOK. Protection & Security.

Text books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles, Wiley, 8/e.
2. W. Stallings, Operating Systems Internals and Design Principles, Prentice Hall (2013) 7th ed.
3. Andrew S. Tanenbaum, Operating Systems: Design and Implementation, Pearson (2006)
4. Andrew S. Tanenbaum, Modern Operating Systems, Pearson 5th edition (2022)

Reference books:

1. Dhamdhere, D.M., Operating Systems: A Concept Based Approach, McGraw Hill (2008) 2nd ed.
2. Flynn, I.M. and McHoes, A.M., Understanding Operating Systems, Thomson (2007).
3. C. Crowley, Operating Systems. A Design-Oriented Approach, Tata McGraw-Hill Publishing Company Limited, 1998.

Others:

1. https://onlinecourses.nptel.ac.in/noc20_cs04/preview
2. https://archive.nptel.ac.in/courses/106/105/106105214/

CS254	**Data Communication and Computer Networking** **3-0-3** **Core**
Pre-requisite	Programming language Effective from: 12/4/2023
Course Objectives	1. To introduce Internet, LAN, WAN topologies and how to evaluate their performance, Students learn important applications' implementation details: Web applications, E-mail, File Sharing etc. 2. To explore and learn TCP/IP protocol stack and functions of various layers and Students understand and use working principles of TCP protocol and UDP protocol in detail. 3. To learn how to design IP addressing for various networks, routing concepts, and routing protocols (RIP, OSPF, BGP). 4. To get thorough knowledge of Ethernet protocols CSMA/CD, Aloha, Slotted Aloha. 5. To explore and understand the importance of various protocols DHCP, ARP, DNS in networking.
Course Outcomes	At the end of the course, student will be able to: 1. Understand basic networking architectures and TCP/IP protocol stack details. 2. Learn network application architectures and socket programming to understand and implement a variety of network applications. 3. Design of transport layer protocols related to network reliability, flow control, and congestion control. 4. Explore network layer protocols (IP and routing) and their role in setting up intranets, and Internet. 5. Experiment with link layer protocols with sample network topologies.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2		2				
CO2	2		3		2	2	
CO3			3				
CO4		2	2	2	2		
CO5		1	1				
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3		1		1			1
CO4		1		1			1
CO5		1					

Detailed Syllabus

Unit-1:
Fundamental of Data communication and Computer Networks: Data Communication system components, data flow, types of connection, topologies, Internet, Network Edge, Core, Protocol layers and their service models, Network Performance metrics
Unit-2:
Application Layer: Principles of network applications, TCP/UDP socket programming, Web and HTTP, FTP, E-Mail and Internet, DNS and P2P applications
Unit-3: Transport Layer: Transport layer services, UDP, TCP state machines, headers, connection management, reliability protocols, flow control, congestion protocols.
Unit-4:
Network Layer: Introduction, Datagram Networks, Router, IP addressing, subnetting, classless IP addressing, DHCP, Routing Algorithms: link state, distance vector, and Internet routing protocols: RIP, OSPF, BGP, iBGP, eBGP, and routing issues handling
Unit-5: Link Layer: Introduction to Link Layer, Error Detection and Correction, MAC protocols (CSMA/CD, Aloha, Slotted Aloha), LANs, VLANs, forwarding algorithms, ARP protocol

Text books:

1. Computer Networking: A Top-Down Approach – by James Kurose (Author), Keith Ross (Author)

Reference books:

1. Data Communications and Networking - by Forouzan
2. Advanced Network Simulations Simplified - Dr Anil Kumar Rangisetti (Author), Packt

CS255	**Embedded Systems and IoT**	**3-2-4**	**Core**
Pre-requisite	Digital Logic Design, Computer Organization		Effective from: 12/4/2023
Course Objectives	1. To introduce students to the programming and design principles of embedded systems. 2. To provide an understanding of the integration of sensors, actuators, and communication protocols in the design of IoT applications. 3. To equip students with practical knowledge to develop and analyze embedded systems and IoT applications. 4. To prepare students for careers in the field of embedded systems and IoT.		
Course Outcomes	1. Students will be able to design, implement and test embedded systems using programming languages and various hardware platforms. 2. Students will be able to develop applications using various communication protocols commonly used in IoT applications. 2. Write assembly language programs for microprocessors and microcontroller 3. Students will be able to analyze, design and debug complex embedded systems. 4. Students will be able to develop and present technical reports on embedded systems and IoT applications		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Introduction to embedded systems and IoT, Embedded systems and their applications, IoT and its architecture, Sensors, actuators and communication protocols used in IoT, Embedded systems hardware and software platforms overview, Embedded Programming Fundamentals, C Programming, Data types, arithmetic and logical operations, Control structures: decisions and loops, Functions, arrays, and pointers, Introduction to embedded systems programming using C language

Microcontrollers, Microcontroller architecture, Atmel AVR and ARM processors, Programming strategies for embedded systems Examples of Embedded Systems, Applications of embedded systems, Examples of embedded systems in Robotics, Embedded systems in biomedical devices, Case studies of real-world embedded systems

FPGA Programming: Introduction to FPGA and its Architecture, HDL (Hardware Description Language) basics, FPGA Design Flow and Development Tools, FPGA Programming using Verilog/VHDL

Raspberry Pi Programming: Introduction to Raspberry Pi and its Architecture, Raspberry Pi GPIOs and their usage, Programming GPIOs in Python, Raspberry Pi and IoT.

IoT Applications, Cloud computing Overview, Cloud Platforms and services in IoT, Building and deploying IoT applications, Introduction to NodeRED and Real-Time IoT data processing Security and Privacy in IoT Applications, Security Threats in IoT, Assurance techniques and mechanisms for IoT security, Privacy risks in IoT and mitigation Strategies, Legal Considerations in IoT

Reference books:

1. "Embedded Systems: Introduction to Arm Cortex-M Microcontrollers" by Jonathan Valvano
2. "Real-Time Embedded Components and Systems" by Sam Siewert and John Pratt
3. "Internet of Things: A Hands-On Approach" by Arshdeep Bahga and Vijay Madisetti
4. "Exploring Raspberry Pi: Interfacing to the Real World with Embedded Linux" by Derek Molloy
5. "FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version" by Pong P. Chu

CS256	**Automata Theory and Compiler Design Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	1. To demonstrate the interplay between different models and formal languages. 2. Employ finite state machines to solve problems in computing. 3. Classify machines by their power to recognize the languages. 4. Explain deterministic and non-deterministic machines. 5. Emphasize concepts learned in lexical analysis, syntax analysis, semantic analysis, intermediate code generation, and type checking through programming exercises. 6. Understand language translation by designing a complete translator for a mini language.		

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2	2	2	1	2		
CO2	3	3	2	1	3		
CO3	3	3	3	1	3		
CO4	3	3	3	2	3		
CO5	3	3	3	2	3		
CO6	3	3	3	2	3		
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	2	1	1	2	2
CO2	1	2	3	2	1	3	3
CO3	1	2	3	2	1	3	3
CO4	1	2	3	2	1	3	3
CO5	1	2	3	2	1	3	3
CO6	1	2	3	2	1	3	3

List of Experiments

1. Finite Automata and Regular Languages:
Experiment 1: Implement DFA for recognizing a specific regular language.
Experiment 2: Convert NFA to DFA using subset construction.
Experiment 3: Minimize a given DFA.
Experiment 4: Simulate DFA to check membership of a string in the language.
2. Regular Expressions and Finite Automata: Experiment 5: Convert a given regular expression to an NFA using Thompson's construction.
Experiment 6: Convert a given regular expression directly to DFA.
3. Context-Free Grammars (CFGs) and Pushdown Automata (PDAs):
Experiment 7: Parse and generate strings from a given CFG.
Experiment 8: Convert CFG to Chomsky Normal Form (CNF).
Experiment 9: Use CYK algorithm to check string membership in CFG.
Experiment 10: Design and implement a PDA for a context-free language.
Experiment 11: Simulate PDA for string recognition.
4. Lexical Analysis:
Experiment 12: Design a lexical analyzer using finite state machines and implement using Lex/Flex.
Experiment 13: Tokenize input strings based on regular expressions.
5. Syntax Analysis:
Experiment 14: Implement a recursive descent parser.
Experiment 15: Implement an LL(1) parser.
Experiment 16: Construct LR parsing tables and implement an SLR parser.
6. Syntax-Directed Translation:
Experiment 17: Implement syntax-directed definitions for simple expressions.
Experiment 18: Translate arithmetic expressions into postfix notation.
7. Intermediate Code Generation:
Experiment 19: Generate three-address code for arithmetic expressions.
Experiment 20: Implement intermediate code generator for control flow statements.

Reference books:

1. J Hopcroft, JD Ullman and R Motwani, Introduction to Automata Theory, Languages and Computation, Third Edition, Pearson, 2008.
2. Alfred Aho, Ravi Sethi and Jeffrey D Ullman, Compilers Principles, Techniques and Tools, Pearson Education, 2003.
3. M Sipser, Introduction to the Theory of Computation, Second Edition, Thomson, 2005.
4. Lewis H.P. and Papadimition C.H. Elements of Theory of Computation, Prentice Hall of India, Fourth Edition, 2007. 4. S. Arora and B. Barak, Computational Complexity: A Modern Approach, Cambridge University Press, 2009.
5. C. H. Papadimitriou, Computational Complexity, Addison Wesley Publishing Company, 1994.
6. D. C. Kozen, Theory of Computation, Springer, 2006. 7. D. S. Garey and G. Johnson, Computers and Intractability: A Guide to the Theory of NP- Completeness, Freeman, New York, 1979.
7. Alfred Aho, Ravi Sethi and Jeffrey D Ullman, Compilers Principles, Techniques and Tools, Pearson Education, 2003.
8. Levine J.R, Mason T, Brown D, Lex & Yacc, O'Reilly Associates, 1992. 9. Allen I. Holub, Compiler Design in C, Prentice Hall, 2003.
10. Kamala Krithivasan and R Rama, Introduction to Formal Languages, Automata Theory and Computation, Pearson Education, 2009.

CS257	Operating Systems Practice 0-3-2	Core
Pre-requisite	Computer Architecture, Algorithms, Data Structures	Effective from: 12/4/2023
Course Objectives	1. To understand the purpose, structure and functions of operating systems. 2. To understand what a process is and how processes are synchronized and scheduled. 3. To understand different approaches to memory management. 4. To understand the structure and organization of the file system. 5. To know how to design a simple operating system.	
Course Outcomes	At the end of the course, student will be able to: 1. Understand the architecture and functionalities of modern OS. 2. Identify the problems related to process management and synchronization and apply learned methods to solve basic problems. 3. Apply semaphores and monitors for classical and real-world synchronization scenarios. 4. Comprehend the cause and effect related to deadlocks and to analyse them related to common circumstances in operating systems. 5. Perceive the use of virtual memory in modern operating systems and the structure of the most common file-systems.	

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2	1	1	1	1		
CO2	3	3	2	2	1		
CO3	1	2	1	2	2		
CO4	1	2	1	2	1		
CO5	2	1	1	1	1		
CO	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1						1	1
CO2						2	1
CO3						2	2
CO4						3	3
CO5						2	2

Detailed Syllabus

List of Experiments:

1. Implement Non-Preemptive CPU scheduling algorithms: First Come First Server (FCFS) Scheduler, Shortest Job First (SJF) Scheduler, Longest Job First (LJF) Scheduler, Highest Response Ratio Next (HRRN) Scheduler, Non-Preemptive Priority Scheduler

2. Implement Preemptive CPU scheduling algorithms: Shortest Remaining Time First (SRTF) Scheduler, Longest Remaining Time First (LRTF) Scheduler, Priority Scheduler, Round Robin Scheduler

3. Peterson's Solution (for Critical Section Problem)

4. Producer Consumer Problem (using Sleep () and Wakeup() System Calls)

5. Implement the following problems using Semaphores: Producer Consumer Problem, Readers-Writers Problem, Dining Philosophers Problem

6. Implement the following problems using Monitors: Producer Consumer Problem, Readers-Writers Problem, Dining Philosophers Problem

7. Implement the Banker's Algorithm for Deadlock Avoidance

8. Implement the following Disk Scheduling Algorithms: FCFS, SSTF (Shortest Seek Time First), SCAN, Circular SCAN, LOOK, Circular LOOK

Text books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles, Wiley, 8/e.
2. W. Stallings, Operating Systems Internals and Design Principles, Prentice Hall (2013) 7th ed.
3. Andrew S. Tanenbaum, Operating Systems: Design and Implementation, Pearson (2006)
4. Andrew S. Tanenbaum, Modern Operating Systems, Pearson 5th edition (2022)

Reference books:

1. Dhamdhere, D.M., Operating Systems: A Concept Based Approach, McGraw Hill (2008) 2nd ed.
2. Flynn, I.M. and McHoes, A.M., Understanding Operating Systems, Thomson (2007).
3. C. Crowley, Operating Systems. A Design-Oriented Approach, Tata McGraw-Hill Publishing Company Limited, 1998.

Others:

1. https://onlinecourses.nptel.ac.in/noc20_cs04/preview
2. https://archive.nptel.ac.in/courses/106/105/106105214/

CS258	**Computer Networks Practice** **0-3-2** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To introduce Internet, LAN, WAN topologies and how to evaluate their performance, Students learn important applications' implementation details: Web applications, E-mail, File Sharing etc. 2. To explore and learn TCP/IP protocol stack and functions of various layers and Students understand and use working principles of TCP protocol and UDP protocol in detail. 3. To learn how to design IP addressing for various networks, routing concepts, and routing protocols (RIP, OSPF, BGP). 4. To get thorough knowledge of Ethernet protocols CSMA/CD, Aloha, Slotted Aloha. 5. To explore and understand the importance of various protocols DHCP, ARP, DNS in networking.
Course Outcomes	At the end of the course, student will be able to: 1. Understand basic networking architectures and TCP/IP protocol stack details. 2. Learn network application architectures and socket programming to understand and implement a variety of network applications. 3. Design of transport layer protocols related to network reliability, flow control, and congestion control. 4. Explore network layer protocols (IP and routing) and their role in setting up intranets, and Internet. 5. Experiment with link layer protocols with sample network topologies.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	2		2				
CO2	2		3		2	2	
CO3			3				
CO4		2	2	2	2		
CO5		1	1				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3		1		1			1
CO4		1		1			1
CO5		1					

Practice Common Experiments

1. Explore and practice Unix Networking commands
2. Exploring and practice a variety of network analysis tools
3. Learn Network socket programming (TCP/UDP)
4. Design and implementing networking applications
5. Setting up networking topologies using network simulators
6. Analyzing TCP/IP protocols applications performance evaluations using Network Simulators

Text books:

1. Computer Networking: A Top-Down Approach – by James Kurose (Author), Keith Ross (Author)

Reference books:

1. Data Communications and Networking - by Forouzan
2. Advanced Network Simulations Simplified - Dr Anil Kumar Rangisetti (Author), Packt

Semester V

DS301	Managerial Economics and Financial Accountancy 3-0-3 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	Upon completion of this course, the students should be able to: 1. To introduce various concepts of economics and familiarize with the students the importance and application of managerial economics approached in business decisions. 2. To develop an ability to identify, formulate, and solve engineering problems by applying the subject knowledge of Production and cost analysis. 3. To have a better understanding of various Market Structure for wise decision making in Business. 4. To analyze the impact of Macro Economic factors on Business Environment. 5. Evaluate and interpret the financial statements to make informed decisions.

Detailed Syllabus

Nature, scope and methods of managerial economics, The economics of a business and managerial decision making, Ten Principles of Economics, The circular flow model, Production possibility frontier, micro vs macroeconomics.

The concept of demand and supply Analysis: Demand Analysis: Concept, types and determinants of demand, Law of Demand and its limitations Elasticity of Demand, Types and Measurement Supply Analysis: Concept, types and determinants of supply, elasticity of supply.

Applications of supply, demand and elasticity Market equilibrium of supply, demand and price: the managerial challenge. Demand Estimation and Forecasting: Factors governing Demand Forecasting, Methods of Demand Forecasting (Survey Methods, Expert Opinion, Delphi method, Controlled Experiment, Judgemental Approach, and Time Series Analysis etc.).

Reference books:

1. . Managerial Economics: Economic Tools for Today's Decision Makers, Paul G Keat and Philip K Y Young, Pearson education latest edition.

2. Principles of Economics by N G Mankiw, Cengage Learning, latest edition.

3. Managerial Economics: Principles and Worldwide Applications- Dominick Salvatore, Oxford University Press.

4. Financial Accounting for Management - Ramachandran N, Ram Kumar Kakani, ed Pearson Education.

5. Managerial Economics D N Dwivedi, PHI Publication.

CS301	**Machine Learning 3-0-3 Core**
Pre-requisite	Probability and Statistics, Effective from: 12/4/2023 Basics in programming
Course Outcomes	1. To familiarize the students with traditional and modern learning paradigms with their applications in the real-world systems 2. To instill adaptation of human training for development of intelligent machines 3. To inculcate modeling of a real-world practical problem in a machine learning domain 4. To introduce modern artificial neural networks and to develop an understanding of the deep learning techniques 5. To explore some open areas of research and explore directions for their possible solutions

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	2												1	1
CO2	2			1		2		1				1	2	1
CO3	3	2	3	2	2								3	3
CO4			2	1									2	2
CO5	2	3	2	2					1				1	3

Detailed Syllabus

Introduction to machine learning: learning systems, classification, clustering, regression, separability of problems; introduction to learning paradigms: supervised, unsupervised, semi-supervised, active, reinforcement with examples; cross-validation; performance evaluation metrics for classification and clustering; curse of dimensionality, feature selection, reduction and expansion, computation of Eigen co-ordinates and principle component analysis.

Recognition systems and design cycle, Non-linearly separable problems: solutions through Cover's theorem with examples, parametric learning mechanisms like Maximum likelihood, expectation maximisation, aposteriori probabilities, Instance-based learning, Lazy learning with K-nearest neighbour, Eager learning with basis functions, non-parametric learning using support vector machines (SVMs).

Artificial neural networks: Analogy of biological neural network with artificial neural network; Perceptron learning; gradient descent algorithm; multi-layer perceptrons; back-propagation algorithm; activation functions, delta rule, learning curves: overfitting and underfitting of models; Hebbian learning, self-organising feature map, radial basis function neural networks.

Deep neural networks: Introduction and advent of deep learning paradigm, solutions to vanishing and exploding gradient problems, regularisation, activation functions for deep learning, deep feed forward network, convolutional neural network (CNN), pretrained CNN models.

Attention network, generative models like auto-encoders and adversarial learning, recurrent neural networks, problem solving through deep learning and open areas of research, applications.

Text books:

1. T. M. Mitchell, Machine Learning, McGraw-Hill, 1997.
2. S. Haykin, Neural Networks: A Comprehensive Foundation. Prentice-Hall of India, 2007.

Reference books:

1. R. O. Duda, P.E. Hart, D. G. Stork, Pattern Classification, John Wiley, 2001.
2. I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.

CS302	Database Management Systems	3-0-3	Core
Pre-requisite	Data Structures and Algorithms		Effective from: 12/4/2023
Course Objectives	1. Understand functional components of the DBMS. 2. Devise queries using Relational Algebra, Relational Calculus and SQL. 3. Design database schema. 4. Develop E-R model. 5. Evaluate and optimize queries 6. Analyze transaction processing, concurrency control and recovery techniques.		

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Introduction to DBMS: Historical perspective, File Versus a DBMS, Advantages of DBMS, Describing and storing data in DBMS, Architecture of a DBMS, Different Data Models;
Entity Relationship (ER) model: Features of ER model, conceptual design using ER model, design for large enterprises; Relational model–structure and operations, Integrity constraints over relations;
Query languages: Relational Algebra, Relational Calculus and SQL– Queries, Constraints, Form of SQL query, UNION, INTERSECT and EXCEPT, Nested queries, Aggregate Operators, Null values, Complex Integrity constraints in SQL, triggers and Embedded SQL;
Database Design: Mapping ER model to Relational form; Functional Dependency–Closer of functional dependencies, closer of attributes, canonical cover and Properties of Decompositions; Normalization process – 1NF, 2NF, 3NF and BCNF; Multivalued dependency– Closer properties of Multivalued dependency and 4NF; Join dependency– PJNF, Decomposition Algorithms;
Transaction Management: ACID properties, transactions, schedules and concurrent execution of transactions; Concurrency control – lock-based protocol, Serializability, recoverability, dealing with deadlocks and Concurrency control without locking;
Query Processing: Overview of Query Evaluation, operator evaluation; Algorithms for relational operations– Selection operation, General selection condition, Projection operation, Join operation, set operation and aggregate operation, Evaluation of relational operations; Query optimization: Alternative plans, functions of query optimizer, translating SQL queries into relational algebra, estimating the cost of a plan, relational algebra equivalences, and other approaches to query optimization;
Database Recovery: Failure classification, Recovery and atomicity, Log-based recovery shadow paging and Advanced Recovery Techniques: Security and Authorization: Access control, direct access control and Mandatory access control, Role of DBA, Application development.

Reference books:

1. Elamsri, Navathe, Somayajulu and Gupta, Fundamentals of Database Systems, 6th Edition, Pearson Education, 2011.

2. Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems, 3nd Edition, McGraw Hill, 2003.

3. Silberschatz, Korth and Sudharshan, Database System Concepts, 6th Edition, McGraw Hill, 2010.

2. Authors, Title of the Text Book, Name of the Publisher, Year of publication, Edition

Others:

1. MIT Open Course Ware: https://ocw.mit.edu/courses/6-830-database-systems-fall-2010/

CS303	**Software Engineering 3-0-3**	**Core**
Pre-requisite	DBMS, problem solving, CoA	Effective from: 12/4/2023
Course Objectives	1. To impart skill and knowledge in application software development process, tools and techniques.	
Course Outcomes	1. Decompose the software project in various phases of a lifecycle and activities 2. Choose a process for user requirements and draw (using BPMN) 3. Evaluate various Analysis and Design alternatives. 4. Assessment of software product quality 5. Able to plan and schedule software project management activities	

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1			1			1	
CO2		1	1	1			
CO3		1	1				
CO4					1		
CO5			2				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						1	2
CO2				1			2
CO3				1		1	2
CO4				1	1	1	2
CO5				3		2	2

Detailed Syllabus

Introduction to Software Engineering amp; Life cycle models: The evolving role of software, Changing Nature of Software, Software developer skills, and full stack development and technologies. Process models: The waterfall model, Spiral, Incremental process models, Evolutionary process models, Agile process models, Incremental development, Test driven development.

Software Requirements engineering: Requirement classification, Functional and non-functional requirements, User classification and requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management, documentation -UML. Models: Business process modelling(BPM), Context Models, Behavioral models, Data models, Object models, structured methods.

Software Design Engineering: Design concepts, process, the design model, pattern based software design. Creating an architectural design: software architecture, Data design-ER- diagrams, Architectural styles and patterns, Architectural Design, assessing alternative architectural designs, mapping data flow into software architecture. Modeling component-level design: Designing class-based components, conducting component-level design, object constraint language, designing conventional components.

Software Quality engineering: Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging. Object-oriented testing strategies and techniques. Software quality assurance models and metrics. Software metrics: Software structure complexity, Frame work for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Various metrics for Process and Products.

Software Project management Project planning, scheduling, Gantt chart, Critical path, CPM/PERT. Risk management. Estimation models. Quality Management: Quality concepts, Software quality assurance, Software inspection amp; reviews, Statistical Software Quality Assurance, Software reliability, The ISO quality standards.

Usability Engineering: UI and UX design, principles, Analysis, design steps, usability evaluation methods and techniques.

Text books:

1. Software engineering A practitioner's Approach, Roger S Pressman, sixth edition McGraw Hill International Edition. Software Engineering, Ian Sommerville, seventh edition, Pearson education.

Reference books:

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.

2. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008

3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2005 Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press. 4. Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.

5. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley amp;Sons Ltd.

CS304	Data Mining 3-0-3	Core
Pre-requisite	Nil	Effective from: 12/4/2023
Course Outcomes	At the end of the course, a student can : 1. Analyze Algorithms for frequent item sets. 2. Analyze Algorithms for sequential patterns. 3. Determine patterns from time series data. 4. Develop algorithms for graph patterns. 5. Apply Graph mining algorithms to Web Mining.	

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Knowledge Discovery Process. Data Pre-processing Techniques. Data Mining Tasks. Basic concepts of Association Rule Mining, Frequent Item set mining, Mining various kinds of association rules, Sequential Pattern Mining concepts, primitives, scalable methods; Transactional Patterns and other temporal based frequent patterns, Mining Time series Data, Periodicity Analysis for time related sequence data, Trend analysis, Similarity search in Time-series analysis; Graph Mining, Mining frequent subgraphs, finding clusters in large graphs; Web Mining, Mining the web page layout structure, mining web link structure, Automatic classification of web documents and web usage mining

Text books:

1. Jiawei Han and M Kamber, Data Mining Concepts and techniques, Morgan Kaufmann Publishers In, 2022; Fourth Edition.

2. Chris Chatfield, The Analysis of Time Series: An Introduction, Chapman & Hall/CRC, 2003, Sixth Edition.

3. Bing Liu, Web Data Mining, Springer, 2011, Second Edition.

Reference books:

1. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar, Introduction to Data Mining, Pearson, 2018, Second Edition.

2. G. Dong and J Pei, Sequence Data Mining, Springer, 2007, First Edition.

CS305	**Machine Learning Practice 0-3-2**	**Core**
Pre-requisite	Nil	Effective from: 12/4/2023
Course Outcomes	1. To give hands-on experience on data science using real-world data. 2. To develop implementation skills for decision making using advanced tools and algorithms 3. To introduce practical deep learning skillset to the students 4. To expertise the students in using advanced machine learning libraries and tune them for AI-based products	

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	**1**	1	3	1					1				**3**	**2**
CO2	**1**	**2**	**2**	**1**	**3**				1		1	1	**3**	**2**
CO3	1	**2**	**3**	**1**	**2**				1			1	**3**	**3**
CO4		3	**3**	**3**					1		1	1	**3**	**3**

List of Experiments

1. Basic programming to read and analyse structured data finding the number of features, dimensions, resizing of unstructured data, normalisation, scaling, augmentation etc.
2. Data analysis using various plots (like box, scatter, line, histogram, bar charts, etc.) and analysis of statistical distance and divergence measures on the data to draw preliminary conclusions.
3. Data preprocessing exercises like the extrapolation, interpolation, label encoding, one-hot-encoding, redundancy elimination, splitting into train-test sets, etc.
4. Extraction of discriminative features and information from very high dimensional data using principal component analysis, linear discriminant analysis, locally linear embedding, t-distributed Stochastic Neighbor Embedding.
5. Implementation of Naive Bayes classifier as a parametric learning principle
6. Implementation of supervised classification and regression using linear and logistic regression computation of statistical performance indices like precision, recall, F1-score, etc.
7. Implementation of K-NN and K-means algorithms for classification and clustering, respectively, in realisation of supervised against unsupervised learning.
8. Implementation of decision trees and random forest classifiers on standard dataset for decision making
9. Implementation of support vector machines with various kernel functions for multi-class classification problem
10. Implementation of artificial neural networks, gradient descent algorithm, activation functions, loss functions using standard libraries.

Additional Experiments:

1. Implementation of convolutional neural networks with different functional layers (pooling, batch normalisation, etc.) in extraction and decision making out of standard datasets.

Text books:

1. Müller, Andreas C., and Sarah Guido. Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.", 2016.

Reference books:

1. Raschka, Sebastian, and Vahid Mirjalili. Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2. Packt Publishing Ltd, 2019.

Others:

1. Machine learning A-Z by Udemy https://www.udemy.com/course/machinelearning/
2. Machine learning by Andrew Ng on coursera https://www.coursera.org/learn/machine-learning

CS306	**Database Management Systems Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	1. Design and Implement a database Schema 2. Devise Queries using DDL, DML, DCL and TCL Commands 3. Develop application programs using PL/SQL 4. Design and implement a project using embedded SQL and GUI 5. Apply Modified components for performance tuning in open source software		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

List of Experiments

Familiarization of Oracle RDBMS, SQL*Plus and Oracle developer,
SQL: query-structure; DDL-create, alter, drop, rename and Truncate; DML-select, insert, update, delete and lock; Set operations- union, intersection and except; join; Aggregate Operations- group-by and having; nested sub-queries and views; DCL-grant and revoke, TCL-Commit, save point, rollback and set transaction.
PL/SQL: Environment, block structure, variables, operators, data types, control structures; Cursors structures- Implicit and Explicit; Bulk statements- Bulk collect into and forall; Exception handling Compilation and Run-time, user-defined; Stored procedures- creation options, pass-by-value and functions-pass-by-value; Packages-package specification, body, package creation and usage; Triggers Data definition language triggers, Data manipulation triggers, Compound triggers and trigger restrictions;
Large objects: CLOB, NCLOB, BLOB and BFILE; Implementation of applications using GUI; group project;

Reference books:

1. James, Paul and Weinberg, Andy Oppel, "SQL: The Complete Reference", 3rd Edition, McGraw Hill, 2011.

CS307	**Software Engineering Practice**	**0-3-2**	**Core**
Pre-requisite	Problem solving, algorithms, DBMS		Effective from: 12/4/2023
Course Objectives	1. To impart skill and knowledge in application software development process, tools and techniques.		
Course Outcomes	1. Decompose the software project in various phases of a lifecycle and activities 2. Choose a process for user requirements and draw (using BPMN) 3. Evaluate various Analysis and Design alternatives. 4. Assessment of software product quality 5. Able to plan and schedule software project management activities		

List of Experiments

Students should take one real-world problem and start implementing the following
1. Software requirement specification: Draw a BPMN diagram using any open source tool for a Pizza order/delivery/payment process. Note: Download tool, use swim lanes to draw and explain.
Time : 15 days.
2. Project Management: Draw a Gantt-chart for final exam preparation for all courses considering no of hours per day you study, and chapters left over for study. Also show any topic dependencies in the chart.
Time : 20 Days
3. Software development experience: Form a 4/5 member team. Take one of the following problem and develop application software and present Requirement specification, UX design, database design, Modular design. Work distribution among the team members should be specified.
A) IIITKDM examination scheduling and auto-room allocation
B) IIITKDM hostel management
C) Any other Problem statement.
Time : 3 months

Text books:

1. Software engineering A practitioner's Approach, Roger S Pressman, sixth edition McGraw Hill International Edition. Software Engineering, Ian Sommerville, seventh edition, Pearson education.

Reference books:

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
2. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2005 Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press. 4. Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.
5. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley amp;Sons Ltd.

Semester VI

DS351	**Entrepreneurship and Management Functions** **3-0-3** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To introduce the concept of entrepreneurship and its importance in promoting economic growth and development. 2. To provide an understanding of the various entrepreneurship skills and competencies required to start and run a successful business venture. 3. To facilitate the development of a business plan and implementation strategies that will allow students to apply their knowledge in a practical manner. 4. To highlight the importance of management functions such as planning, organizing, staffing, directing and controlling in a business venture.
Course Outcomes	1. Students will be able to identify and evaluate business opportunities by conducting market research and analyzing industry trends. 2. Students will be able to develop a comprehensive business plan that includes a feasible financial plan, marketing strategy, and operations plan. 3. Students will have an understanding of the various legal and regulatory issues associated with starting and running a business. 4. Students will have an understanding of the various management functions and their role in the success of a business venture. 5. Students will be able to communicate effectively by presenting their business plan and strategies in a clear and concise manner.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1			1			1	
CO2		1	1	1			
CO3		1	1				
CO4					1		
CO5			2				

CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1						1	2
CO2				1			2
CO3				1		1	2
CO4				1	1	1	2
CO5				3		2	2

Detailed Syllabus

Introduction to entrepreneurship and management functions: Definition and importance of entrepreneurship, Role of entrepreneurship in economic growth and development, Overview of management functions in a business venture

Entrepreneurship Skills and Competencies, Creativity and Innovation, Opportunity recognition and evaluation, Market research and analysis, Financial management, Marketing strategy and planning, Operations and supply chain management, Legal and regulatory issues

Developing a Business Plan, Components of a business plan, Feasibility analysis, Financial planning and forecasting, Marketing planning and execution, Operations and management planning, Implementation strategies and timeline, Evaluation and control

Management Functions, Definition and importance of management functions in a business venture, Planning and organizing, Staffing and recruitment, Directing and motivating, Controlling and evaluation

Reference books:

1. Entrepreneurship: Successfully Launching New Ventures, Bruce R. Barringer and R. Duane Ireland
2. "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses" by Eric Ries
3. "Small Business Management: An Entrepreneur's Guidebook" by Mary Jane Byrd and Leon Megginson
4. "Entrepreneurial Finance: Fundamentals of Financial Planning and Management for Small Business" by J. Chris Leach and Ronald W. Melicher
5. "Principles of Management" by Mason A. Carpenter, Talya Bauer, and Berrin Erdogan

CS351	High Performance and GPU Computing	3-0-3	Core
Pre-requisite	Nil	Effective from: 12/4/2023	
Course Objectives	1. To conceptualize the basics of organizational and architectural issues of HPC 2. Understand High Performance Computing (HPC) system architectures and various computational models. 3. Learn basics of CUDA programming. 4. Understand the mechanisms for evaluating the suitability of different HPC solutions to solving scientific problems.		
Course Outcomes	Students will be able to: 1. Ability to understand the design principles of HPC 2. Get familiarize with popular parallel programming paradigms. 3. Exhibit knowledge to measure, assess and analyse the performance of HPC applications. 4. Understand the role of communication, workload and resource management in an HPC management software. 5. Design compute intensive applications on HPC platform.		

Detailed Syllabus

Parallel Programming Platforms (Models of Parallel Computing Systems): Motivating Parallelism and Scope of Parallel Computing; Implicit Parallelism: Trends in Microprocessor Architectures, communication Model of Parallel Processing Platforms, Introduction to Multi-core Architecture - -An overview of Parallel Computing Platforms (SIMD and MIMD systems, Clusters); Physical Organization of Parallel Platforms (Architecture of an Ideal Parallel Computer Interconnection Networks for Parallel Computers, Network Topologies, Evaluating Static and Dynamic Interconnection Networks, Cache Coherence in Multiprocessor Systems, Communication Costs in Parallel Machines, Routing Mechanisms for Interconnection Networks (Store and Forward Routing /Cut-through Routing Methods- Implementation aspects on different parallel systems. Principle of Parallel algorithm: Data and Task Parallelism; Various Decomposition Methods /Techniques; Task Dependency Graphs, Granularity, Concurrency, and Task-Interaction, Processes and Mapping, , Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing (Static and Dynamic Load Balancing; Design Parallel Algorithm: (Partitioning (Decomposition Methods); Communication; Agglomeration (load balance computations); Mapping)
One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather All-to-All Personalized Communication; Performance aspects on Store-Forward and Cut- through Routing; Performance aspects on various Parallel Computing Models. Analytical Model of Parallel Programs: Optimization and Performance aspects of Sequential programs; Sources of Overhead in Parallel Programs, Performance Metrics for Parallel Systems (Speed Up, Performance, and Scalability analysis; Amdahl's Law, and Gustafson Law), Scalability of Parallel Systems;
Programming Shared Address Space Platforms – An Overview Implementation; An Overview of OpenMP The OpenMP programming Model (Concurrent Tasks, Synchronization Constructs, Data Handling); OpenMP-Environment Variables; POSIX threads (Pthreads), Synchronization primitives, Threads-mutex and condition variables, Synchronization constructs, Threads Versus OpenMP; Intel Threading Building Blocks (Intel TBB) – Express Parallelism in C++ program; Containers, Scalable Memory Allocation; Mutual Exclusion; Task Scheduler; Implementation and Performance of Matrix Computations using OpenMP, Pthreads and IntelTBB. Programming Using the Message-Passing Paradigm – MPI : An overview of the Message Passing Building blocks (Sending and Receiving Operations); An Overview of Message Passing Interface- MPI Point-Point, Collective communication and Computation library calls an MPI I/O library Calls; Performance and implementation aspects of algorithms based on Numerical Computations and Non-Numerical Computations using MPI library calls

Parallel Programming and Algorithms Implementation - Performance Aspects : Matrix Computations Matrix Vector Multiplication (Row wise – one- and two- dimensional Partitioning); Different parallel Matrix-Matrix Multiplication algorithms; Cannon algorithm Solving system of Linear Equations; Sparse Matrix Computations; Parallel Programming and Algorithms Implementation - Performance Aspects - Sorting algorithms Issues in Sorting on Parallel Computers; Sorting Networks; Bubble Sort and its Variants Quicksort; Parallelizing Quicksort; Parallel Programming and Algorithms Implementation - Performance Aspects; All-pairs of Shortest Paths Algorithms; Sequential and Parallel Search Algorithms; Depth-First Search Algorithms; Best-First Search Algorithms; Parallel algorithms for - Graph Computations; Graph Partitioning Algorithms; Graph Colouring Algorithms

Parallel Algorithms Design and Implementation aspects on Multi-Core Systems with GPU accelerators; History of GPUs; An Overview of GPU Programming (OpenCL Programming Paradigms; CUDA enabled NVIDIA GPUs) ; An Overview of GPU Architecture (Hardware /Software); An Overview of CUDA enabled NVIDIA GPUs, CUDA Libraries for Numerical Computations; Algorithms for Matrix Computations using CUDA, OpenCL, OpenACC and Performance Analysis; An Overview of OpenCL : OpenACC (High Level GPU Programming APIs), CUDA APIs, CUDA Libraries for Numerical Computations; Performance of CPU-GPU based programs for numerical computations using openACC; An Overview of OpenCL; The OpenCL – Heterogeneous Programming; OpenCL Libraries, The OpenCL Memory Model, Execution Model; Platform and Devices – Programming APIs using OpenCL. Programming Concepts and Design Algorithms for Parallel Computing systems with GPU Accelerators for Numerical Computations; Programming paradigms using MPI, OpenMP, PThreads with GPU Accelerators (CUDA enabled NVIDIA GPUs); Mixed Programming for Matrix Computations Performance Issues; Other Advance Concepts.

Reference books:

1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar : Introduction to Parallel Computing, Second Edition Pearson Education – 2007
2. Michael J. Quinn, Parallel Programming in C with MPI and OpenMP McGraw-Hill International Ed (2003)
3. Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley (2011)
4. Tom White, The Hadoop Definitive Guide, O'RELLY Media 2009 Study Material
5. Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufman Publishers, Elsevier (2011)
6. Barbara Chapman, Gabriele Jost, And Ruud Van der Pass Using OpenMP Portable Shared Memory Parallel Programming MIT Press (2008)
7. Rohit Chandra, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, Ramesh Menon, Parallel Programming in OpenMP, Academic Press (2001)

CS352	Computer Vision 3-0-3 Core
Pre-requisite	Programming, Probability and Statistics, Machine learning Effective from: 12/4/2023
Course Outcomes	1. To introduce the basics of digital image processing and early computer vision techniques 2. To introduce the major ideas, methods, and techniques of computer vision and pattern recognition 3. To inculcate the idea of extracting meaningful information from images 4. To suggest a design of a computer vision system for a specific problem4. To motivate development of major computer vision based application to cater to the societal needs 5. To explore some open areas in computer vision research

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2
CO1	2									1			2	1
CO2	2	2	3	1	1								3	2
CO3		2	2	1	2								1	2
CO4	1		3	1									1	3
CO5		3		3			2		1			1		2

Detailed Syllabus

Course Overview and Motivation; Introduction to Image Formation, Capture and Representation; Linear Filtering, Correlation, Convolution: Edge, Blobs, Corner Detection; Scale Space and Scale Selection; SIFT, SURF; HoG, LBP, etc.Bag-of-words, VLAD.

RANSAC, Hough transform; Pyramid Matching; Optical Flow. Review of Deep Learning, Multi-layer Perceptrons, Backpropagation Introduction to CNNs; Evolution of CNN Architectures: AlexNet, ZFNet, VGG, InceptionNets, ResNets, DenseNets Visualization of Kernels; Backprop-to-image/Deconvolution Methods; Deep Dream, Hallucination, Neural Style Transfer; CAM,Grad-CAM, Grad-CAM++; Recent Methods (IG, Segment-IG, SmoothGrad).

CNNs for Recognition and Verification (Siamese Networks, Triplet Loss, Contrastive Loss, Ranking Loss); CNNs for Detection: Background of Object Detection, R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD, RetinaNet; CNNs for Segmentation: FCN, SegNet, U-Net, Mask-RCNN Review of RNNs; CNN + RNN Models for Video Understanding: Spatio-temporal Models, Action/Activity Recognition Introduction to Attention Models in Vision; Vision and Language: Image Captioning, Visual QA, Visual Dialog; Spatial Transformers; Transformer Networks.

Review of (Popular) Deep Generative Models: GANs, VAEs; Other Generative Models: PixelRNNs, NADE, Normalizing Flows, etc Applications: Image Editing, Inpainting, Superresolution, 3D Object Generation, Security; Variants: CycleGANs, Progressive GANs, StackGANs, Pix2Pix, etc.

Recent Trends: Zero-shot, One-shot, Few-shot Learning; Self-supervised Learning; Reinforcement Learning in Vision; Other Recent Topics and Applications.

Reference books:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, 2016.
2. Michael Nielsen, Neural Networks and Deep Learning, 2016.
3. Yoshua Bengio, Learning Deep Architectures for AI, 2009.
4. Richard Szeliski, Computer Vision: Algorithms and Applications, 2010.
5. Simon Prince, Computer Vision: Models, Learning, and Inference, 2012.
6. David Forsyth, Jean Ponce, Computer Vision: A Modern Approach, 2002.

Others:

https://archive.nptel.ac.in/courses/106/106/106106224/

CS353	**HPC and GPU Computing Practice**	**0-3-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. Understand the basic concepts of HPC and the different hardware and software components of an HPC system. 2. Learn how to design and implement parallel algorithms that can efficiently utilize the computing resources of an HPC system. 3. Develop hands-on experience with programming languages and tools commonly used in HPC, such as C/C++, OpenMP, CUDA, and MPI. 4. Gain knowledge about the different application areas of HPC. 5. Analyze and optimize the performance of parallel algorithms on HPC systems by measuring key metrics.		
Course Outcomes	Students will be able to: 1. Design and develop parallel algorithms for HPC applications. 2. Implement and optimize GPU-based algorithms using CUDA and OpenCL programming models. 3. Measure and analyze the performance of parallel algorithms using profiling and benchmarking tools. 4. Design and implement parallel algorithms for solving scientific and engineering problems using HPC systems and GPUs 5. Apply parallel programming concepts and techniques to optimize the performance of real-world applications on multi-core CPUs and GPUs.		

List of Experiments

The aim is to write codes and demonstrate performance of Code with respect to parallel algorithm on target platforms such as Multi-Core Systems with Ubuntu OS and GPUs or MPI based cluster with GPU Accelerators on each node. You should use PThreads; OpenMP-GPU or Intel TBB; MPI-GPU; mixed programming (MPI-OpenMP, MPI Threads) and Hadoop MapReduce or SPARK-Python framework. Measure Sustained Performance for Algorithms implementation based on Numerical Computations (Matrix Computations) and Non Numerical Computations (Sorting, Search, Graph Computations) using (i) Compiler Optimisations (ii) Vendor Supplied or Open-Source Software Math Kernel Library. For each question, you may require to write a code using different programming paradigms. The total marks assigned for your submitted code will depend upon the many factors such as a) check for valid input, b) comment cards to be written; c) correctness of the results, d) test the results for large problem size with varying no. of processes (Threads) as per question; f) clean exit of the code for in-valid input data and print of the output data, g) explanation of your code justifying the use of APIs or functions and h) modularity

1. Implementation of Sequential (Serial) programs for Matrix Computations; Use of Compiler Technology and extract performance for Dense Matrix Computations; Use of Vendor supplied or open source Software Numerical Math Libraries (BLAS-I, BLAS-II, BLAS-II, DGEMM) and demonstrate Performance of matrix computations; Basics of Programming on Shared Address Space Arch. Using OpenMP

2. Basic Programs on Matrix Computations using Pthreads, OpenMP, Intel TBB, and MPI; Basic Programs on Matrix-Matrix multiplication – Cannon's, and Fox algorithm using MPI

3. Dense and Sparse Matrix Computations with Vendor Supplied Math Library using Pthreads, OpenMP, Intel TBB, and MPI and Performance Issues

4. Sorting Algorithms – Bitonic, Odd-even transposition, Quicksort, Bucket Sort by MPI Implementation – Performance Issues

5. Minimum spanning tree and Shortest Path Algorithms – MPI Implementation; All-pairs shortest paths – Dijkstra's algorithm, Floyd's algorithm – MPI Implementation

6. Maximal independent set and Graph Colouring Algorithms for sparse graphs; MPI Implementation for large Graphs

7. Basic Programs on Dense Matrix Computations using Pthreads, OpenMP, MPI, CUDA NVIDIA-GPU, Programs using CUDA enabled NVIDAI Libraries; Iterative Methods to solve Matrix System of Linear Equations by OpenMP-GPU

8. Programs using CUDA enabled NVIDAI Libraries; and Iterative Methods solve Matrix System of Linear Equations by MPI, MPI-CUDA –NVIDIA GPU, Basic Matrix Programs using Vendor Supplied or Open Source Software Libraries such as BLAS – Math Kernels.

9. Heterogeneous Programming on Systems with different accelerators using OpenCL; The OpenCL Kernel, The OpenCL Models - Memory, Execution, Platform and Devices; OpenCL Execution Environment, An Overview of OpenCL API; Performance of Matrix Computations Algorithms using OpenCL using Math Kernel Libraries

10. Basic Programs using Hadoop MapReduce and Spark-Python for Word Count, Matrix into Vector Multiplication; Basic Programs using Spark-Python. Large Size Data Intensive programs for Hadoop MapReduce with HDFS, Spark-Python-HDFS

Reference books:

1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar : Introduction to Parallel Computing, Second Edition Pearson Education – 2007

2. Michael J. Quinn, Parallel Programming in C with MPI and OpenMP McGraw-Hill International Ed (2003)

3. Jason Sanders, Edward Kandrot, CUDA By Example – An Introduction to General-Purpose GPU Programming, Addison Wesley (2011)

4. Tom White, The Hadoop Definitive Guide, O'RELLY Media 2009 Study Material

5. Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufman Publishers, Elsevier (2011)

6. Barbara Chapman, Gabriele Jost, And Ruud Van der Pass Using OpenMP Portable Shared Memory Parallel Programming MIT Press (2008)

7. Rohit Chandra, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, Ramesh Menon, Parallel Programming in OpenMP, Academic Press (2001)

CS354	**Computer Vision Practice 0-3-2**	**Core**
Pre-requisite	Nil	Effective from: 12/4/2023
Course Outcomes	1. To familiarize with basic image processing techniques through a hands-on training 2. To develop basic skills in extracting meaningful information from images after enhancing them through filters 3. To develop advanced skills in object detection and captioning of images 4. To expertise in industry-oriented deep learning techniques for advanced image analysis and monitoring	

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	**1**	1	3	1					1				**3**	**2**
CO2	**1**	**2**	**2**	**1**	**3**				1		1	1	**3**	**2**
CO3	1	**2**	**3**	**1**	**2**				1			1	**3**	**3**
CO4		3	**3**	**3**					1		1	1	**3**	**3**

List of Experiments

1. Basics of digital image processing: resizing, contrasting, cropping, flipping, rotation, zooming, changing colour contours, visualizations of dimensions from image matrices, etc.
2. Spatial operations on images: Contrast stretching, threshold function, intensity transformations, negating an image, fourier spectrum analysis, gamma correction, histogram processing and equalisation, bit-plane slicing.
3. Application of filters: blurring, sharpening, smoothing, enhancing, edge detection through spatial filters, Canny edge detection, Hough transform, convolution.
4. Frequency domain transformation: extraction of temporal features, frequency filters, discrete fourier transform, entropy measurement, etc.
5. Implementation of artificial neural networks using patterns from images and challenges.
6. Convolution neural networks for automatic extraction of spatial features, layer-wise operations, activation layers, basic decision making, image captioning.
7. Image context analysis through use of recurrent neural networks and analysis.
8. Attention neural network model in analysis of important portions of an image.
9. Implementation of auto-encoders in image denoising 10. Implementation of auto-regressive models like GANs, deep belief networks, etc.

Additional Experiments:

1. Implementation of semantic segmentation techniques like U-net, Mast RCNN, etc. for object detection.

Text books:

1. Vasilev, Ivan, et al. Python Deep Learning: Exploring deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow. Packt Publishing Ltd, 2019.

Reference books:

1. Koul, Anirudh, Siddha Ganju, and Meher Kasam. Practical Deep Learning for Cloud, Mobile, and Edge: Real-World AI Computer-Vision Projects Using Python, Keras TensorFlow. O'Reilly Media, 2019.

Others:

1. Python for Computer Vision with OpenCV by Udemy https://www.udemy.com/course/python-for-computer-vision-with-opencv-and-deep-learning.
2. Computer Vision Basics — Coursera https://www.coursera.org/learn/computer-vision-basics.
3. https://www.coursera.org/learn/introduction-computer-vision-watson-opencv.

ID351	Product Design Practice 0-3-2 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To enable students to understand the principles of product design 2. To provide students with practical experience in product design process, including ideation, prototyping, and testing 3. To equip students with the necessary knowledge and skills to design products that meet user needs and specifications 4. To prepare students for a career in product design or related fields
Course Outcomes	1. Students will be able to apply design thinking principles to product design process 2. Students will be able to generate ideas, develop concepts, and create prototypes that meet project objectives and user needs 3. Students will be able to test and evaluate product prototypes to determine their suitability for mass production and use in the real world 4. Students will be able to demonstrate communication skills through the creation of effective design briefs and presentations

List of Experiments

1. Introduction to product design and design thinking.
2. Developing skills in observation, user research, and problem identification.
3. Idea Generation and Concept Development
4. Developing and evaluating product concepts.
5. Sketching and Rendering, Creating design sketches for products using CAD software.
6. 3D Modeling and Prototyping
7. Prototyping and Testing
8. Presenting and Communicating Design Solutions

Reference books:

1. "The Design of Everyday Things" by Don Norman
2. "Observing the User Experience" by Elizabeth Goodman, Mike Kuniavsky, and Andrea Moed
3. "Design Thinking: Integrating Innovation, Customer Experience, and Brand Value" by Thomas Lockwood and Edgar Papke
4. "Product Design and Development" by Karl Ulrich and Steven Eppinger 5. "Sketching: Product Design Presentation" by Koos Eissen and Roselien Steur

Semester VII

CS498	**Project I** **3-0-3** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. Synthesize and apply prior knowledge to design and implement solutions to open-ended computational problems while considering multiple realistic constraints. 2. Design and Develop the software with SE practices and standards 3. Analyze Database, Network and Application Design methods 4. Evaluate the various validation and verification methods and practice CASE tools for solving SE CASE Studies 5. Analyze professional issues, including ethical, legal and security issues, related to computing projects.

CS496	**Seminar** **0-3-2** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Demonstrate an understanding of the importance of technical writing in Computer Science and Engineering 2. Develop effective technical writing skills such as writing precise and clear instructions, reports and proposals 3. Effectively use computer-based tools for writing, research and formatting 4. Identify and apply a variety of technical writing styles 5. To develop research and presentation skills
Course Outcomes	1. Analyze the selected topic, organize the content and communicate to audience in an effective manner 2. Write standard technical documents like reports, proposals and articles 3. Use appropriate citation methods for technical documents 4. Create stunning visual aids (e.g. graphics and diagrams) 5. Employ well-tested editing and formatting techniques

Detailed Syllabus

Technical Writing and its scope Classroom Activity: Group discussions with brief presentations; overview of Technical writing genres (Reports, Proposals, Instructions and User Manuals). Classroom Activity: Industry reports and proposal analysis; Business and Technical writing principles Classroom Activity: Analyzing principles of effective technical Writing; Language Standards and Editing Techniques Classroom Activity: Peer review and editing assignments; Technical Writing tools (Software and Platforms) Classroom Activity: Real-time practice with software tools; Graphics and Visual Aids Classroom Activity: Practical exercises in creating graphical representations;

Semester VIII

CS499	**Project II 3-0-10 Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. Synthesize and apply prior knowledge to design and implement solutions to open-ended computational problems while considering multiple realistic constraints. 2. Design and Develop the software with SE practices and standards 3. Analyze Database, Network and Application Design methods 4. Evaluate the various validation and verification methods and practice CASE tools for solving SE CASE Studies 5. Analyze professional issues, including ethical, legal and security issues, related to computing projects.

CS497	**Internship 0-0-3 Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. The ability to apply the theoretical knowledge gained in the classroom to real-world engineering problems and projects, and gain practical experience in solving them. 2. The ability to work in teams and communicate effectively with team members, supervisors, and clients, and develop interpersonal skills that are essential for success in the engineering profession. 3. The opportunity to gain insights into the engineering industry's practices and culture, and develop an understanding of the professional and ethical standards that are expected of an engineer. 4. The opportunity to build professional networks with industry experts and peers, and learn from their experiences and perspectives, which can help in career advancement and professional growth. 5.

Design Electives

XXXX	**Design History** **2-0-2** **Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. Understand the evolution and application of the concept of Design in everyday life of people. 2. Appreciate its role in national and international economic and social systems. 3. Analyze the emerging designs from a societal perspective.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1			2			1	1
CO2			1			2	1
CO3			3			1	1

CO	PO8	PO9	PO1	PO1	PO1	PSO1	PSO2
CO1							
CO2							
CO3							

Detailed Syllabus

Definition of Design; Origin of designers; Historical context of design and designers.
Designers and designed products: Art, design and technology - Select International and Indian designers.
Industrial Revolution: Mass production, Birth of Modern architecture, International Style, The modern home.
Craft and Design: Type forms; William Morris and Arts and Craft Movement; Shantiniketan.
Design movements: Art Nuoveau; Art Deco, Werkbund; Bauhaus; De Stijl. Changing values
Information Revolution: Impact of technology, industrialization and globalization on design: kitsch, pastiche, 'retro'; Shopping malls.
Design Studies: Materials and techniques; Chinese ceramics; Typology; Content analysis:
Anthropology / sociology; Nationalist and global trends in Design; Nationalist Design; Global trends and global identity; Nostalgia, Heritage and Design;

Text books:

1. Conway Hazel, Design History – A Students' Handbook, Routledge: London, 1987.

Reference books:

1. Raizman David, History of Modern Design, Graphics and Products since the Industrial Revolution. Laurence King Publishing: London, 2003
2. Walker John. A, Design History and History of Design. Pluto Press: London, 2003.
3. Woodham Jonathan M, Twentieth Century Design, Oxford University Press: Oxford, 2003

XXXX	Sociology of Design 2-0-2 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. Students develop empathy, and contribute to social causes, leading to a better understanding of societal needs and issues. 2. Students develop leadership skills, communication skills, teamwork, and critical thinking abilities, which are essential for personal growth and career development 3. Students learn about their social responsibility towards nation, and develop a sense of civic duty

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1			3			2	1
CO2			2			2	1
CO3			1			2	1

CO	PO8	PO9	PO1	PO1	PO1	PSO1	PSO2
CO1		2					
CO2		2					
CO3		2					

Detailed Syllabus

Basics concepts of sociology (behavior, interaction, language).
Historical evolution of Societies (Agrarian, Industrial, Digital) and current human and organizational contexts in which engineers and other professionals work, Personal and corporate social responsibility amp;amp; ethics.
Relationship between people (age, gender, cultures) and technology - Social and psychological dimensions of technological change, Technology amp;amp; Work, Co-operative Work; Coordinative Practices, Ethno methodology, Critical Systems Heuristics.

Text books:

1. Manuel Castells (1996); The Rise of Network Society.
2. Herbert Blumer (1986); Symbolic Interactionism: Perspective and Method.

Reference books:

1. Herkert, J. (ed.), Social, Ethical, and Policy Implications of Engineering: Selected Readings. New York, NY: IEEE Press, 2000.
2. Heath, C. and Luff, P. (2000); Technology in Action, Cambridge: Cambridge University Press.
3. Werner Ulrich (1983), Critical Systems Heuristics, John Wiley.

XXXX	Earth, Environment and Design	2-0-2	Core
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	1. Understand the basics of environmental issues relevant to the development of modern technology. 2. Know the principles behind modern eco-system. 3. Understand the concept of sustainable development. 3. Have awareness about various environmental policies and allied socio-economic issues.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1			2			2	3
CO2							3
CO3			2			1	3
CO4			2			1	3

CO	PO8	PO9	PO1	PO1	PO1	PSO1	PSO2
CO1	1				1		
CO2					1		
CO3					1		
CO4					1		

Detailed Syllabus

Introduction to environment and ecology – Ecosystems – Principles concepts, components and function, Atmospheric, aquatic and terrestrial ecosystems – Biogeochemical cycles and limiting factor concepts –Impacts of natural and human activities on ecosystems.

Environmental policies, acts and standards – Sustainable development and environmental impact assessment – Institutional frame work and procedures for EIA Methods for impact identification-matrices – Networks and Check lists – Environmental settings, indices and indicators.

Prediction and assessment of the impacts on air, water, land, noise and biological environments – Assessment of impacts of the cultural, socioeconomic and eco-sensitive environments.

Mitigation measures, economic evaluation – Public participation and design making –Preparation of Environmental statement.

Text books:

1. Rubin. E. S, Introduction to Engineering and the Environment, McGraw Hill, 2000.

2. Masters. G. M., Introduction to Environmental Engineering Science, Prentice Hall,1997.

Reference books:

1. Henry. J. G, and Heike, G. W, Environmental Science Engineering, Prentice Hall International, 1996.

Dhameja. S. K, Environmental Engineering and Management, S. K. Kataria and Sons, 1999.

Shyam Divan and Armin Rosancranz, Environmental Law and Policy in India, Cases, Materials and Statutes, Oxford University Press, 2001.

Humanities Electives

XXXX	**Technical and Professional Ethics for Engineers**	**2-0-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	1. Understand the difference between morals and ethics. 2. Understand the concept of Professional Ethics. 3. Improve their integrity and morality. 3. Differentiate between good and bad practices		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							

CO	PO8	PO9	PO1	PO1	PO1	PSO1	PSO2
CO1	3	1			2		
CO2	3	1			2		
CO3	3	1			2		
CO4	3	1			2		
CO5	3	1			2		

Detailed Syllabus

Professionalism and Ethics: Profession and occupation, Qualities of a professional practitioner, Variety of ethics and moral issues, moral dilemmas; Kohlberg's theory - Gilligan's theory of moral development - consensus and controversy. Values- concept of intrinsic good, instrumental good and universal good. Kant's theory of good action and formula for universal law of action. Codes of ethics for engineers: need and scope of a code of ethics; Ethics and Law

Understanding Ethical Problems: ethical theories – utilitarianism, cost-benefit analysis, Duty ethics - Right ethics and virtue ethics. Applications for various case studies.

Ethical Problem Solving Techniques: issues-factual, conceptual and moral; Bribery and acceptance of gifts; Line drawing and flow charting methods for solving conflict problem.

Risk, Safety and Accidents: Safety and risk, types of risk, types of accidents and how to avoid accidents. Rights and Responsibilities of an Engineer: Professional responsibility, professional right and whistle blowing. Ethical Issues in Engineering Practice: environmental ethics, computer ethics, ethics and research.

Text books:

1. Charles D. Fleddermann, "Engineering Ethics", Pearson Education / Prentice Hall, New Jersey,2004.

Reference books:

1. Charles E Harris, Michael S. Protchard and Michael J Rabins, "Engineering Ethics – Concepts and Cases", Wadsworth Thompson Leatning, United States, 2000. 2. Velasquez. M. G, Business Ethics and Cases, 5 Edn, Prentice Hall, 2002.

3. Sekha. R.C, Ethical Choices in Business Response, Sage Publication, 2002.

4. Mike Martin and Roland Schinzinger, Ethics in Engineering, McGraw Hill, 1996.

XXXX	**Universal Human Values 2-0-2 Core**
Pre-requisite	Nil Effective from: 12/4/2023
Course Outcomes	1. Understand basic universal human values. 2. Understand the concepts of harmony, health and prosperity. 3. Analyze and understand the value of human relations with family and society.
	4. Understand concepts such as harmony in nature and existence in relation to ethics will be introduced.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1	PO1	PO1	PSO1	PSO2
CO1						1		2	1					
CO2						1	1							
CO3						3	2		1					
CO4									1					
CO5												1		

Detailed Syllabus

Course Introduction - Need, Basic Guidelines, Content and Process for Value Education Purpose and motivation for the course, recapitulation from Universal Human Values Self-Exploration–what is it? - Its content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration Continuous Happiness and Prosperity- A look at basic Human Aspirations Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario Method to fulfil the above human aspirations: understanding and living in harmony at various levels. Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking.

Understanding Harmony in the Human Being - Harmony in Myself! Understanding human being as a co-existence of the sentient 'I' and the material 'Body' Understanding the needs of Self ('I') and 'Body' - happiness and physical facility Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer) Understanding the characteristics and activities of 'I' and harmony in 'I' Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail Programs to ensure Sanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease. Understanding Harmony in the Family and Society- Harmony in Human- Human Relationship Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship Understanding the meaning of Trust; Difference between intention and competence Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.

Understanding Harmony in the Nature and Existence - Whole existence as Coexistence. Understanding the harmony in the Nature Interconnectedness and mutual fulfilment among the four orders of nature recyclability and self-regulation in nature Understanding Existence as Co-existence of mutually interacting units in all- pervasive space Holistic perception of harmony at all levels of existence. Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

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

Text books:

1. 1. R R Gaur, R Asthana, G P Bagaria, “A Foundation Course in Human Values and Professional Ethics”, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93- 87034-47-1.
2. R R Gaur, R Asthana, G P Bagaria, “Teachers’ Manual for A Foundation Course in Human Values and Professional Ethics”, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-53-2.

Reference books:

1. 1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amar kantak, 1999.
2. A. N. Tripathi, “Human Values”, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. Mohandas Karamchand Gandhi “The Story of My Experiments with Truth”.
5. E. FSchumacher. “Small is Beautiful”.
6. Slow is Beautiful –Cecile Andrews.
7. J C Kumarappa “Economy of Permanence”.
8. Pandit Sunderlal “Bharat Mein Angreji Raj”.
9. Dharampal, “Rediscovering India”.
10. Mohandas K. Gandhi, “Hind Swaraj or Indian Home Rule”.
11. India Wins Freedom - Maulana Abdul Kalam Azad.
12. Vivekananda - Romain Rolland (English).
13. Gandhi - Romain Rolland (English).

XXXX	**History of Science and Technology in India**	**2-0-2**	**Core**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	At the end of the course, students will be able to: 1. Understand historical development of Science and Technology in India starting from Ancient Period. 2. Understand the Science and Technology of Ancient India. 3. Understand development of Science and Technology in Medieval Period. 4. Analyze and understand Modern development of Scientific Intuitions, Scientific Temper. 5. Conduct relevant case studies.		

CO-PO/PSO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	3	1	1	2	2	1	1
CO2	2		3		3		
CO3	2	1			1		1
CO4	3	3			3		
CO5	1	1			1	1	
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1					
CO2			2				
CO3		2	1				
CO4							
CO5	1	3					

Detailed Syllabus

Ancient History Science and Technology in the age of Harappa and Indus valley civilizations, Science in Vedic and Post Vedic period, Development of Mathematics, Astronomy, Medicine, etc.
Medieval period Mathematics, Astronomy, Medicine, Textile, Art and Architecture, Irrigation, Materials and metallurgy, etc.
Modern Period Modern Scientific institutions, Scientific temper, Science Popularization movements, Atomic Energy, Space Science, Agriculture, etc.
Case Studies

Reference books:

1. Bernal, J. D. (1969). Science in history: Vol. 1-4.
2. Narlikar, J. V. (2003). The scientific edge: The Indian scientist from vedic to modern times.
3. Habib, I. (2020). Technology in Medieval India, Tulika Books.
4. Mahesh, V. S. (2011). Science and Technology in Ancient India, Centrum Press.
5. Kamlesh, M. (2014). Science and Technology in Colonial India, Aakar Books.

Electives

CS401	**Game Theory**	**3-0-3**	**Elective**
Pre-requisite	Probability and Statistics, Discrete Mathematics		Effective from: 12/4/2023
Course Objectives	1. To introduce concepts of game theory. 2. To represent players, actions and games in proper mathematical notations. 3. To inculcate rational and strategic thinking among students using game theory examples. 4. To encourage students to model the real-world scenarios in terms of games. 5. To find solutions solutions to different game-theory problems systematically.		
Course Outcomes	At the end of the course, student will be able to: 1. Formulate various engineering problems as games. 2. Interpret the strategic interactions in games. 3. Solve the games to determine the Nash Equilibrium. 4. Examine the applicability of game theory concepts to real life scenarios. 5. Choose appropriate methods to determine solutions for games (if any).		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2	1	2	2									
CO2	3	2	1	2	3								3	
CO3	1	1	1	2	2								3	
CO4	3	3	2	3								3		2
CO5	3	3	1	2	1									3

Detailed Syllabus

The Basics of Game Theory: Introduction, overview, uses of game theory, some applications and examples, and formal definitions of: the normal form, payoffs, strategies, pure strategy Nash equilibrium, dominant strategies.

Mixed-Strategy Nash Equilibrium: Pure and mixed strategy Nash equilibria, applicability and methods to find mixed strategy Nash equilibria

Alternate Solution Concepts: Iterative removal of strictly dominated strategies, minimax strategies and the minimax theorem for zero-sum game, correlated equilibria.

Extensive-Form Games: Perfect information games: trees, players assigned to nodes, payoffs, backward Induction, subgame perfect equilibrium, introduction to imperfect-information games, mixed versus behavioral strategies.

Repeated Games: Repeated prisoners' dilemma, finite and infinite repeated games, limited-average versus future-discounted reward, folk theorems, stochastic games and learning. Introduction to Bayesian Games, and Coalitional Games.

Text books:

1. M. J. Osborne and A. Rubinstein, "A Course in Game Theory," MIT Press, 1 st Edition, 1994.
2. R. B. Myerson, "Game Theory: Analysis of Conflict," Harvard University Press, 1st Edition, 1997

References:

1. https://papers.ssrn.com/sol3/papers.cfm?abstractid=1968579.
2. http://www.gtessentials.org.

CS402	Internet of Things	3-0-3	Core
Pre-requisite	Probability and Statistics, Computer Networks, Computer Architecture, and atleast one high level programming language		Effective from: 12/4/2023
Course Objectives	1.To cover the building blocks of Internet of Things (IoTs) and their characteristics. 2. To introduce the reader to the programming aspects of Internet of Things with a view towards rapid prototyping of complex IoT applications. 3. To learn how to design and develop IoT applications 4. To learn how to integrate IoT with cloud computing and big data analytics		
Course Outcomes	On the successful completion of the course, the students would be well-versed with the Concepts of IoT. Specifically, the outcomes of this course are as follows: 1. To understand the real-world importance of IoT. 2. To understand the working of different IoT devices including the protocols, and interfaces. 3. To be able to map the hardware and software constituents of an IoT device. 4. Integrate IoT with cloud computing and big data analytics		

Detailed Syllabus

Introduction: Definition, characteristics, physical design, logical design, enabling technologies, and levels.
Domain Specific IoTs: Introduction, Home Automation, Cities, Environment, Energy, Retail, Agriculture, and Industry.
IoT and M2M: Introduction, M2M, difference between IoT and M2M, SDN and NFV for IoT.
IoT, Cloud Computing, and Fog Computing: The need for the Cloud,conventional cloud architecture, disadvantages, and edge computing, the need for the fog, and variants of fog computing.
IoT Platforms Design Methodology: Introduction,motivation, design methodology, specifications, and logical design, what is an IoT device, Raspberry Pi, Linux on Raspberry Pi, and interfaces.

Text books:

1. Internet of Things: A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Hands On Books Series, August 2014

CS403	Natural Language Processing 3-0-3 Elective
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Understand approaches to syntax and semantics in NLP. 2. Understand approaches to discourse, generation, dialogue and summarization within NLP. 3. Understand current methods for statistical approaches to machine translation. 4. Understand machine learning techniques used in NLP, including hidden Markov models and probabilistic context-free grammars, clustering and unsupervised methods, log-linear and discriminative models, and the EM algorithm as applied within NLP.
Course Outcomes	1. Identify the different linguistic components of natural language. 2. Evaluate a morphological analyser for a given natural language. 3. Apply appropriate parsing techniques necessary for a given language. 4. Design new tagset and a tagger for a given natural language. 5. Develop applications involving natural language.

Detailed Syllabus

Morphology And Part-Of-Speech Processing: Introduction –Regular Expressions and Automata-Non-Deterministic FSAs. Transducers –English Morphology-Finite-State Morphological Parsing -Porter Stemmer -Tokenization-Detection and Correction of Spelling Errors. N-grams –Perplexity -Smoothing -Interpolation -Backoff . Part-of-Speech Tagging –English Word Classes -Tagsets -Rule-Based -HMM -Transformation-Based Tagging -Evaluation and Error Analysis. Hidden Markov and Maximum Entropy Models Speech Processing: Phonetics –Articulatory Phonetics -Phonological Categories -Acoustic Phonetics and Signals -Speech Synthesis–Text Normalization –Phonetic and Acoustic Analysis -Diphone Waveform synthesis –Evaluation-Automatic Speech Recognition – Architecture -Hidden Markov Model to Speech -MFCC vectors -Acoustic Likelihood Computation -Evaluation. Triphones –Discriminative Training -Modeling Variation. Computational Phonology- Finite-State Phonology –Computational Optimality Theory - Syllabification -Learning Phonology and Morphology. Syntax Analysis: Finite-State and Context-Free Grammars -Dependency Grammars. Syntactic Parsing – Ambiguity -Dynamic Programming Parsing Methods –CKY-Earley and Chart Parsing-Partial Parsing-Evaluation. Statistical Parsing – Probabilistic ContextFree Grammars –Probabilistic CKY Parsing of PCFGs –Probabilistic Lexicalized CFGs – Collins Parser – Shallow parsers – Dependency parsing. Semantic and Pragmatic Interpretation: Representation of Meaning –Desirable Properties -Computational Semantics -Word Senses -Relations Between Senses –WordNet -Event Participants-Proposition Bank -Frame Net -–Metaphor. Computational Lexical Semantics –Word Sense Disambiguation-Supervised Word Sense Disambiguation - Dictionary and Thesaurus Methods-Word Similarity -Minimally Supervised WSD - Hyponymy and Other Word Relations -Semantic Role Labeling -Unsupervised Sense Disambiguation. Computational Discourse -Discourse Segmentation - Unsupervised Discourse -Segmentation -Text Coherence -Reference Resolution –Phenomena –Features and algorithms -Pronominal Anaphora Resolution.

Applications: Information Extraction –Named Entity Recognition -Relation Detection and Classification –Temporal and Event Processing -Template-Filling -Biomedical Information Extraction. Question Answering and Summarization - Information Retrieval -Factoid Question Answering -Summarization -Single and Multi-Document Summarization - Focused Summarization -Evaluation. Dialog and Conversational Agents –Properties of Human Conversations -Basic Dialogue Systems

Text books:

1. Jurafsky and Martin, "Speech and Language Processing", Pearson Prentice Hall, Second Edition, 2008.
2. Christopher D. Manning and Hinrich Schütze, "Foundations of Statistical Natural Language Processing", MIT Press, 1999.

Reference books:

1. Stevan Bird, "Natural Language Processing with Python", Shroff, 2009.
2. James Allen, "Natural Language Understanding", Addison Wesley, Second Edition, 2007.

CS404	Information Retrieval 3-0-3 Core
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To introduce the fundamental concepts and trends in information retrieval systems. 2. To familiarize students with the different types of information retrieval systems and their components 3. To empower students to evaluate and select appropriate information retrieval techniques. 4. To instill knowledge of Information retrieval as a research field, its challenges and future research directions. 5. To assess different search models and their basic structure.
Course Outcomes	Students will be able to: 1. To present the basic concepts in information retrieval and more advance techniques of multi-modal based information systems 2. To understand the underlined problems related to information retrieval. 3. To acquire the necessary experience to design and implement real applications using information retrieval systems 4. Analyze evaluation and performance of retrieval techniques.

Detailed Syllabus

Introduction to Information Retrieval: The nature of unstructured and semi-structured text. Inverted index and Boolean queries. Text Indexing, Storage and Compression, Text encoding: tokenization, stemming, stop words, phrases, index optimization. Index compression: lexicon compression and postings lists compression. Gap encoding, gamma codes, Zipf's Law. Index construction. Postings size estimation, merge sort, dynamic indexing, positional indexes, n-gram indexes, real-world issues.

Retrieval Models: Boolean, vector space, TFIDF, Okapi, probabilistic, language modeling, latent semantic indexing. Vector space scoring. The cosine measure. Efficiency considerations. Document length normalization. Relevance feedback and query expansion. Rocchio, Performance Evaluation: Evaluating search engines. User happiness, precision, recall, F-measure. Creating test collections: kappa measure, interjudge agreement.

Text Categorization and Filtering: Introduction to text classification. Naive Bayes models. Spam filtering. Vector space classification using hyperplanes; centroids; k Nearest Neighbors. Support vector machine classifiers. Kernel functions. Boosting, Text Clustering: Clustering versus classification. Partitioning methods. k-means clustering. Mixture of gaussians model. Hierarchical agglomerative clustering. Clustering terms using documents.

Advanced Topics: Summarization, Topic detection and tracking, Personalization, Question answering, Cross language information retrieval, Web Information Retrieval: Hypertext, web crawling, search engines, ranking, link analysis, PageRank, HITS. Retrieving Structured Documents: XML retrieval, semantic web.

Reference books:

1. Introduction to Information Retrieval Manning, Raghavan and Schutze, Cambridge University Press.
2. Modern Information Retrieval Baeza-Yates and Ribeiro-Neto, Addison Wesley, 1999.
3. A comprehensive survey by Ed Greengrass Mining the Web, Soumen Charabarti, Morgan-Kaufmann, 2002.

CS405	Recommender Systems 3-0-3 Elective
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To introduce students to the principles and techniques of recommender systems. 2. To provide students with an understanding of the different types of recommender systems and how they work. 3. To teach students how to evaluate the effectiveness of different recommender systems. 4. Developing basic concepts of recommender system and its usefulness in various domains. 5. To build confidence among the students to apply a recommender system in new domains
Course Outcomes	1. Students will be able to explain the principles and techniques of recommender systems. 2. Students become aware of various issues related to Personalization and Recommendations. 3. Students will be able to evaluate the effectiveness of different recommender systems. 4. Students will be able to recognize the ethical issues surrounding the use of recommender systems and propose solutions to mitigate them. 5. Students become competent to implement a set of well-known recommender systems.

Detailed Syllabus

Introduction to Recommender Systems- Goals, Basic Models of Recommender Systems, Domain Specific Challenges in Recommender System.

Content Based Recommender System-Basic Components of Content based System, Feature Representation, Learning User profile and Filtering, Case study of web page recommendation, Music Recommendation.

Neighborhood Based Collaborative Filtering- Predicting ratings with neighborhood based methods-User based model, Item-based models, similarity measures in neighborhood based approaches, Strength and weakness of user based and item based methods, A unified view of user-based and item-based method, Graph Models for Neighborhood based Methods. Model Based Collaborative Filtering - Decision and Regression Tree, Naive Bayes Collaborative Filtering, Basic Matrix Factorization, Stochastic Gradient Descent, Incorporating User and Item Biases, Integrating Factorization and Neighborhood based Methods.

Evaluating Recommender System- Evaluation Issues with offline recommender system, Segmenting the Ratings in training and testing, Evaluation Metrics - Accuracy, Mean Absolute Error (MAE), Good Item MAE, Good Predicted Item MAE, Coverage, Novelty, Diversity, Precision, Recall, NDCG

Context Sensitive Recommender System-Contextual Modelling, Factorization Machines. Time and Location Sensitive Recommender Systems-Temporal Collaborative Filtering, Recency based model, Discrete Temporal Models (Markovian), Location Aware Recommender System, PoI Recommendations in Social Network. Structural Recommendation in Network-Personalized PageRank, Recommending Friends-Link Prediction, Trust Centric Recommendation-Building Trust Network, Trust Propagation and Aggregation, Popular Trust Based Recommender Systems (TidalTrust, MoleTrust, TrustWalker Algorithms), Social-Tagging Recommendations. Session based Recommendation using Deep Learning Techniques, Multi-Criteria Recommender Systems, Recommender Systems in Tourism Industry

Advanced Recommender System: Federated Learning, Federated Recommender System, Active Learning in Recommender Systems, Long Tail Issues in Recommendations, Diversity centric personalized recommender systems, Ethical Issues-Privacy concerns, Fairness and transparency issues, Ethical considerations in designing, implementing, and deploying recommender systems, Final Project.

Text books:

1. Charu C. Aggarwal. Recommender Systems: The Textbook (1st. ed.). Springer, USA, 2016.

Reference books:

1. Francesco Ricci, Lior Rokach, Bracha Shapira. Recommender Systems Handbook (3rd ed.). Springer US, 2022.

CS406	**Blockchain**	**3-0-3**	**Elective**
Pre-requisite	Cryptography		Effective from: 12/4/2023
Course Objectives	1. Develop proficiency in essential Crypto Primitives: Public Key Encryption, Digital Signatures, Hash functions, and their protocol constructions. 2. Understand Blockchain Fundamentals, including Cryptocurrency and principles of distributed systems. 3. Explore Consensus Algorithms, focusing on both Permissionless and Permissioned approaches. 4. Dive into Bitcoin technology, including wallet functionalities, Bitcoin Improvement Proposals (BIPs), and processes for transactions and investments. 5. Introduce Ethereum technology, covering transaction mechanics, the role of Ethereum Virtual Machine (EVM), and practical skills in creating and executing smart contracts using Solidity.		
Course Outcomes	At the end of the course, student will be able to: 1. Mastering Encryption for secure communication, Hash functions for data integrity, and Digital Signatures for authentication. Comprehensive understanding of the usability of zero-knowledge proof systems. 2. Proficiency in Blockchain components, transaction details, and roles of key players. 3. Expertise in various Permissionless and Permissioned Consensus algorithms. 4. Competence in Bitcoin operations, including transaction creation and management of crypto wallets. 5. Capability in Ethereum and Smart Contract Development using the Ethereum Virtual Machine (EVM).		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Module - I: Basic Crypto Primitives

Public Key Cryptography - RSA encryption scheme, Hash Functions - Merkle Damgard construction, Merkle Root Tree, Digital Signatures - RSA Signature scheme, Schnorr Signature scheme, (EC)DSA, Zero Knowledge Proof System.

Module - II: Blockchain

Introduction: What is Blockchain, Cryptocurrency to Blockchain - A Historical Journey, Distributed Systems Overview, Cryptocurrency.

Elements of a Blockchain: Block Header, The structure of a block and block header, The genesis block Mining, Task of miners, Transaction Organization, Transaction life cycle, Types of transaction, Transaction fee, Transaction malleability, Transaction pools.

Module - III: Consensus Algorithms

Permissionless Consensus: Proof of Work (PoW), Proof of Stack, Proof of Burn, Proof of Elapsed Time.

Permissioned Consensus: State Machine Replication, Distributed State Machines and Consensus, Crash Fault Tolerance, CFT Consensus, Paxos, Byzantine Fault Tolerance, BFT Consensus, Practical Byzantine Fault Tolerance (PBFT).

Module - IV: Bitcoin

The bitcoin network: Wallets, Payments: Bitcoin investment and buying and selling bitcoins, Bitcoin installation, Bitcoin programming and the command-line interface, Bitcoin improvement proposals (BIPs), Bitcoin limitations.

Module - V: Ethereum

Ethereum Introduction: Ethereum blockchain, Transactions, Smart Contracts creation, Message call transaction, Elements of the Ethereum blockchain, Ethereum Virtual Machine (EVM), Pre-compiled contracts, Accounts, Blocks, Transaction and Block validation mechanism, Introduction to Solidity.

Text books:

1. Imran Bashir, “Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more”, 3rd Edition, Packt Publishing, 2020,
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, “Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction”, Princeton University Press (July 19, 2016)

Online Courses:

1. NPTEL Course on Blockchain and its Applications by Prof. Sandip Chakraborty, Prof. Shamik Sural, https://onlinecourses.nptel.ac.in/noc22_cs44/preview.
2. NPTEL Course on Introduction to Blockchain Technology Applications by Prof. Sandeep Shukla, https://archive.nptel.ac.in/courses/106/104/106104220/#.

CS407	**Object Oriented Programming 3-0-3 Core**
Pre-requisite	C programming Effective from: 12/4/2023
Course Objectives	1. To learn importance of OOP approaches over procedural approaches. 2. To learn OOP features such as Classes, Objects, Member Functions, Data Abstraction, Data Hiding, Inheritance and Polymorphism. 3. To learn how to implement generic data structures and algorithms To learn how to handle run time errors and debugging applications To learn how to use OOP features for implementing real time applications. 4. To learn how to implement generic data structures and algorithms To learn how to handle run time errors and debugging applications. 5. To learn how to use OOP features for implementing real time applications.
Course Outcomes	At the end of the course, student will be able to: 1. Explaining OOP features through CPP. 2. Building basic CPP programs by following OOP principles. 3. Discovering OOP concepts to implement reusable, extendible and easy to use softwares. 4. Evaluating generic algorithms, data structures and applications. 5. Designing and implementing real time simple software.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1		1											
CO2	2		3						3		3			
CO3	2		3		3	2			2		3			
CO4	1	2	3	2	3	2					2		1	
CO5	2	2		2		2			3		3			

Detailed Syllabus

Unit-1 Procedure oriented programming vs OOP, Introduction to object oriented programming and OOP, Basic C++ syntax, data-type, variables,strings, functions, default values in functions, recursion, namespaces, operators, flow control, arrays and pointers

Unit-2 Classes and objects, private, public, constructors, destructors, member data, member functions, inline functions, friend functions, static members, and references,Dynamic memory management in c++, new and delete operators

Unit-3 Introduction to Inheritance: Class hierarchy, derived classes, single inheritance, multiple, multilevel, hybrid inheritance, Virtual base class, constructor and destructor execution, base initialization using derived class constructors

Unit-4 Polymorphism: Binding, Static binding, Dynamic binding, Static polymorphism: Function Overloading, Ambiguity in function overloading,Dynamic polymorphism: Base class pointer, object slicing, late binding, method overriding with virtual functions, pure virtual functions, abstract classes and Operator Overloading.

Unit-5 C++ Advanced features: Exception handling: Try, throw, and catch, exceptions Template: template classes, template functions. Standard Template Libraries, Namespaces Object Oriented Design, design and programming, role of classes

Text books:

1. Object Oriented Programming with C++ by E. Balagurusamy, McGraw-Hill Education (India)

Reference books:

1. C++: The Complete Reference- Schildt, McGraw-Hill Education (India)
2. Mastering C++ - Venugopal, McGraw-Hill Education (India)

CS411	**Social Network Analysis 3-0-3**	**Elective**
Pre-requisite	Nil	Effective from: 12/4/2023
Course Outcomes	1. Understand the importance of social networks and social graphs 2. Enhance analytical skills for analyzing social networking data 3. Develop skills to leverage extended enterprise data 4.Create real-life case studies using social networks	

Detailed Syllabus

Introduction to social network analysis: Graphs – nodes, edges, direct and indirect friends/neighbors, degree and degree distribution, shortest path, cycle, tree, complete graph, bipartite graphs, directed graphs, weighted graphs, adjacency matrix, social interactions and connected components. Technological networks (internet, telephone network, power grids, transportation networks), social networks (facebook, movie collaboration, paper collaboration), information networks (web), biological networks (neural networks, ecological networks).

Network Centrality Measures and Models: Properties of real-world network – degree distribution, clustering coefficient, average path length; Random Graphs – Evolution of random graphs, properties of random graphs, modeling real-world networks with random graphs, Erdos-Renyi model of random graph; Small-world Model – Properties of the Small-world model, modeling real-world networks with the small-world model; Preferential attachment model – Properties of the preferential attachment model, modeling real-world networks with the preferential attachment model.

Random walk-based proximity measures, other graph-based proximity measures. Clustering with random-walk based measures.

Influence and Homophily: Measuring Assortativity, Measuring and modeling Influence, Measuring and modeling Homophily, Distinguishing influence and homophily – shuffle test, edge-reversal test, randomization test; Spread of influence through a network, influence maximization in networks, spread of disease on networks.

Games on networks, game theory strategies, dominant strategies, dominated strategies, pure strategies and mixed strategies, Nash equilibrium, multiple equilibria-coordination games, multiple equilibria-the Hawk-Dove game, mixed strategies, Modeling social network traffic using game theory.

Text books:

1. Reza Zafarani, Mohammad Ali Abbasi, Huan Liu, "Social Media Mining – An Introduction", Cambridge University Press, 2014.

2. David Easley and Jon Kleinberg, "Networks, Crowds, and Markets: Reasoning About a Highly Connected World", Cambridge University Press,2010.

Reference books:

1. Mark Newman, "Networks: An Introduction", Oxford University Press,2010.

2. Hansen, Derek, Ben Sheiderman, Marc Smith, "Analyzing Social Media Networks with NodeXL: Insights from a Connected World", Morgan Kaufmann,2011.

3. Avinash Kaushik, "Web Analytics 2.0: The Art of Online Accountability", Sybex,2009.

CS413	**Unmanned Aerial Vehicle 3-0-3**	**Elective**
Pre-requisite	Flight Mechanics, Classical Control Theory	Effective from: 12/4/2023
Course Objectives	1. To understand the working principles and importance of unmanned aerial vehicles (UAV) in various sectors. 2. To comprehend the design and construction of different types of UAVs. 3. To understand the use of UAVs in military and civilian applications. 4. To analyze the operational status of UAVs.	
Course Outcomes	1. Understand the principles of UAV design and construction. 2. Gain knowledge about the different applications of UAVs. 3. Demonstrate an understanding of the working principles of UAVs. 4. Select the most suitable UAV for a specific application.	

Detailed Syllabus

Definition and history of UAVs Types of UAVs (fixed-wing, rotary-wing, hybrid), UAV components (airframe, propulsion system, guidance and control, payload),

UAV Flight Dynamics, Basic principles of flight, Flight performance, Stability and control of UAVs, Flight control systems

Types of sensors (optical, infrared, radar, etc.), Payload selection and integration Data acquisition and processing, UAV navigation: accelerometers, gyros, GPS. Path planning algorithms: Dubin's curves, way-points, Voronoi partitions. Path following and guidance: Straight line and curve following, vision based guidance.

Future directions and the road ahead, Radio communication principles Communication protocols and frequencies, Ground control stations, Regulations governing UAVs, Safety and ethical considerations, Privacy and security concerns

Civilian and military applications of UAVs, Agricultural, environmental, and disaster management applications, UAVs for surveillance and reconnaissance

Reference books:

1. Randal W. Beard and Timothy W. McLain: Small Unmanned Aircraft:

2. Theory and Practice, Princeton University Press, 2012

3. Kimon P. Valavanis: Advances in Unmanned Aerial Vehicles: State of the Art and the Road to Autonomy, Springer, 2007

CS417	Human Computer Interaction 3-0-3 Elective
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Introduce fundamental theories and concepts of Human Computer Interaction (HCI) 2. Become familiar with the vocabulary associated with sensory and cognitive systems as relevant to task performance by humans 3. Be familiar conventional and non-traditional user interface paradigms 4. Recognize the limits of human performance as they apply to computer operation 5. Understand the social implications of technology and their ethical responsibilities as engineers in the design of technological systems
Course Outcomes	Students will gain: 1. Knowledge of the fundamental aspects of human perception, cognition, and learning. 2. Ability to apply HCI and principles to interaction design. 3. Ability to design certain tools for blind or physically disabled people. 4. Ability to apply models from cognitive psychology to predicting user performance in various human-computer interaction tasks

Detailed Syllabus

Introduction to HCI, Brief History, The human, Computer, Interaction, Human cognitive abilities, perception, memory, attention, and learning.

User Interface Design, Models, Principles, Practices, Direct Manipulation, Overview, Scope, Applications, Interaction of hardware and software in HCI, Interaction design basics, HCI in software process, Design rules.

Cognitive Framework of HCI, Socio-organizational issues and stakeholder requirements, Communication and collaboration models, Task analysis, Dialog notations and design, Perception and representation, Attention and Interface Design, Memory in Interface Design.

Knowledge Representation, User Modeling, Interaction with Natural Languages, Next Generation Interface.

UI Evaluation, Introduction, Cognitive Walkthrough, Heuristic Evaluation, Evaluation with Cognitive Models, Evaluation with Users, Model-based Evaluation.

Reference books:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, —Human Computer Interaction, 3rd Edition, Pearson Education, 2004

2. Gerard Jounghyun Kim. 2015. Human–Computer Interaction Fundamentals and Practice, Taylor Francis Group, LLCCRC Press

3. Human –Computer Interaction, D. R. Olsen, Cengage Learning.

4. Human –Computer Interaction, Smith - Atakan, Cengage Learning.

Others:

1. https://paragnachaliya.in/wp-content/uploads/2017/08/HCI_Alan_Dix.pdf

CS418	Design Patterns 3-0-3 Elective
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. To provide an understanding of design patterns and their importance in software engineering. 2. To enhance the problem-solving skills of students while designing software products. 3. To introduce the different categories of design patterns and their applications in solving real-world software engineering problems. 4. To develop the ability to evaluate and select appropriate design patterns for a given software design problem.
Course Outcomes	1. Outcome 1 Upon successfully completing this course, students will be able to: 1. Explain the concepts of design patterns, their importance and applications in software engineering. 2. Analyze real-world software engineering problems and suggest appropriate design patterns for their resolution. 3. Apply the principles of design patterns to design software systems in a structured and efficient manner. 4. Develop an understanding of the different categories of design patterns and their applications.

Detailed Syllabus

Definition of design patterns and their significance, Types of design patterns with examples, Overview of the differences between design patterns and architectural patterns, Importance of design patterns for software development

Introduction to creational design patterns, Definition of creational patterns, Examples of creational patterns - Singleton, Factory, Abstract Factory, Builder and Prototype, The concept of dependency injection and its use in software design

Introduction to structural design patterns - Adapter, Bridge, Decorator, Facade, Flyweight, and Proxy, Definition of structural patterns, Advantages and disadvantages of structural patterns, Examples of applications of structural patterns

Introduction to behavioral design patterns, Overview of different behavioral patterns - Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method and Visitor, Advantages and disadvantages of behavioral patterns

Examples of applications of behavioral patterns in software engineering, Overview of software architecture, The role of design patterns in software architecture, Integration of design patterns in software architecture, Examples of successful software architectures that utilize design patterns.

Reference books:

1. "Head First Design Patterns" by Eric Freeman, Elisabeth Robson, Bert Bates, and Kathy Sierra

2. "Design Patterns: Elements of Reusable Object-Oriented Software" by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides

3. "Dependency Injection: Principles, Practices, and Patterns" by Steven van Deursen and Mark Seemann

4. "Patterns of Enterprise Application Architecture" by Martin Fowler

5. "Design Patterns Explained: A New Perspective on Object-Oriented Design" by Alan Shalloway and James R. Trott

CS420	**Affective Computing 0-3-2 Elective**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	To introduce students to the concept of affective computing and its applications 2. To provide an understanding of the techniques and tools utilized in developing affective computing systems 3. To enable students to apply affective computing techniques and tools in real-world scenarios
Course Outcomes	Upon completion of this course, students will be able to: 1. Define the concept of affective computing and explain its importance in modern-day technology 2. Apply machine learning techniques to identify and classify human emotions 3. Develop real-world applications that incorporate affective computing techniques and tools 4. Identify the ethical implications of affective computing and propose methods to address these concerns

Detailed Syllabus

Introduction to Affective Computing, Definition and historical background of affective computing, Significance of affective computing in modern day technology, Emotion Analytics, Understanding the importance of emotion recognition and its applications, Machine learning algorithms for emotion recognition
Facial Expression Recognition, Techniques for facial expression recognition, Real world application of facial expression recognition
Speech Emotion Recognition, Techniques for speech emotion recognition, Real world applications of speech emotion recognition, Physiological Sensing, Techniques for physiological sensing, Real world applications of physiological sensing
Affective Computing and Robotics, The role of affective computing in robotics, Examples of affective computing in robots, Ethical Implications of Affective Computing, Ethical concerns surrounding affective computing, Proposed methods to address ethical concerns

Reference books:

1. "Affective Computing" by Rosalind W. Picard
2. "Emotion Recognition Using Facial Expressions" by Sheryl Brahnam and Lakhmi C. Jain
3. "Physiological Computing Systems: A Systematic Review" by Stephen Fairclough
4. "Robot Emotions" by Jordi Vallverdú and David Casacuberta 5. "Ethical and Social Issues in the Information Age" by Joseph Migga Kizza.

CS425	**Software Testing** **3-0-3** **Elective**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	To introduce the concept of software testing and its importance in the software development life cycle. 2. To teach techniques, tools and methodologies for testing software applications. 3. To develop skills for designing and executing tests for software applications. 4. To prepare students to work professionally in software testing.
Course Outcomes	1. Explain the importance of software testing in the software development life cycle. 2. Describe the basic principles and techniques of software testing. 3. Design and implement test plans and test cases for software applications. 4. Use various software testing tools and methodologies. 5. Evaluate software testing results and suggest improvements in the testing process.

Detailed Syllabus

Introduction to Software Testing, Importance of software testing in software development life cycle, Types of tests in software testing, The testing process (planning, designing, execution, reporting), Principles of Software Testing, Why testing is necessary, Verification and validation, Black box and white box testing Types of Testing, Functional testing, Non-functional testing, Acceptance testing, Regression testing, Performance testing, Usability testing, Security testing, Test Design Techniques, Equivalence partitioning, Boundary value analysis

Decision table testing, State transition testing, Test Case Development, Developing test cases and test scenarios, Test data management, Traceability matrix,Test Execution and Reporting

Test execution strategy, Bug tracking and reporting, Test closure, Metrics collection, Automation Testing, Advantages and disadvantages of automation testing, Tools for automation testing

Software Testing Management, Test planning and strategy, Test estimation and scheduling, Test team organization, Practical Exercises, Design and execute test cases for a software application, Test automation using a tool, Test planning and reporting.

Reference books:

1. "Foundations of Software Testing" by Rex Black
2. "Software Testing: Principles and Practices" by Srinivasan Desikan and Gopalaswamy Ramesh
3. "Testing Computer Software" by Cem Kaner, Jack Falk, and Hung Quoc Nguyen
4. "A Practical Guide to Testing in DevOps" by Katrina Clokie
5. "Effective Software Testing: 50 Specific Ways to Improve Your Testing" by Elfriede Dustin, Thom Garrett, and Bernie Gauf.

CS426	**Soft Computing**	**3-0-3**	**Elective**
Pre-requisite	Programming, algorithms		Effective from: 12/4/2023
Course Outcomes	1.Introduce approximate computing as an alternative to traditional algorithms for solutions to computationally complex problems 2. Interpret the range of solutions offered by Fuzzy set theory as compared to crisp set theory 3. Explore the traditional (classical mathematics-based) optimization techniques and understand their limitations 4.Explore single and multi-objective optimisation problems like genetic algorithms, particle swarm optimisation, etc. 5. Design custom architecture to tackle real-world problems through artificial neural networks		

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2
CO1	1	1											1	2
CO2	1	2	1	1									3	2
CO3	2	2		2										1
CO4	2	3	2	3	3								3	3
CO5		3	3	3								2	2	3

Detailed Syllabus

Introduction to Soft Computing: concept of computing systems, "Soft" computing versus "Hard" computing, characteristics of Soft computing, some applications of Soft computing techniques, principle of optimisation taking numerical examples, constrained and unconstrained optimisation problems, single versus multi-variable optimisation problems, direct search methods versus steepest descent methods, local versus global solutions

Fuzzy set theory: Introduction to Fuzzy logic, fuzzy set theory versus crisp set theory, Fuzzy sets and membership functions, Operations on Fuzzy sets, Fuzzy relations, rules, propositions, implications and inferences, Defuzzification techniques, Fuzzy logic controller design. Some applications of Fuzzy logic.

Genetic Algorithms (GA): Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques, Basic GA framework and different GA architectures. GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GAs. Multi-objective Optimization Problem Solving: Concept of multi-objective optimization problems (MOOPs) and issues of solving them, Multi-Objective Evolutionary Algorithm (MOEA), Non-Pareto approaches to solve MOOPs, Pareto-based approaches to solve MOOPs, Some applications with MOEAs, Evolutionary computing techniques like ant-colony optimization, simulated annealing optimization, particle swarm optimisation, etc.

Artificial Neural Networks: Biological neurons and its working, Simulation of biological neurons to problem solving, Different ANNs architectures, Training techniques for ANNs, gradient descent back propagation, Single versus multi layer perceptrons, linearly separable versus non-linearly separable problems, Applications of ANNs to solve some real life problems.

Reference books:

1. Fuzzy Logic with Engineering Applications (3rd Edn.), Timothy J. Ross, Willey, 2010.

2. Foundations of Neural Networks, Fuzzy Systems, and Knowldge Engineering, Nikola K. Kasabov, MIT Press, 1998.

3. Neural Networks and Learning Machines, (3rd Edn.), Simon Haykin, PHI Learning, 2011.

4. An Introduction to Genetic Algorithms, Melanie Mitchell, MIT Press, 2000.

5. Soft computing: fundamentals and applications. D. K. Pratihar, Alpha Science International, Ltd, 2013.

Others:

1. Lecture notes on Soft Computing by Dr. D. Samanta, IIT Kgp, https://cse.iitkgp.ac.in/ dsamanta/courses/sca/resources/slides/01%20Intro.pdf

CS427	Data Science	3-0-3	Elective
Pre-requisite	Nil		Effective from: 12/4/2023
Course Outcomes	Students will be able to: 1. Apply statistical methods to data for inferences. 2. Analyze data using Classification, Graphical and computational methods. 3. Understand Data Wrangling approaches. 4. Perform descriptive analytics and data visualization over massive data.		

Detailed Syllabus

Introduction to data science, data science life cycle, data science process, roles, tools, and technologies, data collection, data wrangling, focusing on techniques for data collection, cleaning, pre-processing, and transformation. Overview of Random variables and probability distributions.

Statistical learning: Assessing model accuracy, Bias-Variance Trade-Off, Descriptive Statistics, Dependent and Independent events; Linear Regression: Simple and multiple linear regressions, regularization, Lasso, Ridge, and Elastic-Net Regression. Comparison of Linear regression with K-nearest neighbours. Logistic Regression, LDA, QDA. PCA and SVD.

Hypothesis Testing, Student's t-test, paired t and U test, correlation and covariance, tests for association; association rules and correlations; hypothesis testing, correlation and causation, ANOVA, and statistical significance.

Exploratory data analysis (EDA), descriptive statistics, data visualization techniques, and identifying patterns and trends, Histograms and frequency polygons, Box-plots, Quartiles, Scatter Plots, Heat Maps. Matrix visualization, Scientific Design Choices in Data Visualization, Higher-dimensional Displays and Special Structures, Visual data mining.

Data Wrangling: Data Acquisition, Data Formats, Imputation, split-apply-combine paradigm. Descriptive Analytics: Data Warehousing and OLAP, Data Summarization, Data de- duplication, Data Visualization using CUBEs. **Text books:**

1. Gareth James Daniela Witten Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, February 11, 2013, web link: www.statlearning.com (1 to 4 Chapters)

2. Mark Gardener, Beginning R : The Statistical Programming Language, Wiley, 2015.

Reference books:

1. Han , Kamber, and J Pei, Data Mining Concepts and Techniques, 3rd edition, Morgan Kaufman, 2012. (Chapter 2 and Chapter 4)

2. C Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.

<table>
<tr><th>CS429</th><th>Graph Theory</th><th>3-0-3</th><th>Elective</th></tr>
<tr><td><b>Pre-requisite</b></td><td colspan="2">Basic Mathematical and Rational Thinking</td><td>Effective from: 12/4/2023</td></tr>
<tr><td><b>Course Objectives</b></td><td colspan="3">1. To introduce to students, the fundamental concepts and techniques of graph theory.
2. To develop ability in students to model real-world problems using graphs and solve them using graph-theoretic algorithms and techniques.
3. To familiarize students with important classes of graphs and their properties.
4. To expose students to various applications of graph theory in computer science, mathematics, networking, etc.
5. To give students an idea of matching in graphs and some applications of matching in day to day life problems.</td></tr>
<tr><td><b>Course Outcomes</b></td><td colspan="3">At the end of the course, student will be able to:
1. Define the basic concepts of graph theory, such as vertices, edges, degree, etc.
2. Contrast the efficiency of different graph algorithms.
3.Classify the graphs comprehensively based on the properties and the traits.
4. Model and solve real-world problems in networks and communications using graph-theoretic techniques, such as graph coloring and matching.
5. Adapt an already existing graph-theory framework to suit the changing needs of the real-world problems.</td></tr>
</table>

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	1		1								1	1
CO2	1	2	2	1	1								2	
CO3		2		2										1
CO4	2	2	3	3	2								1	1
CO5		2	2	2	1							2		1

Detailed Syllabus

Fundamental Concepts: What is a Graph ?, Paths, Cycles and Trails, Vertex Degrees and Counting, Directed Graphs, Basic Properties of Trees, Spanning Trees and Enumeration, Optimization and Trees.

Matchings and Factors: Matchings and Covers, Algorithms and Appplications, Matchings in General Graphs.

Connectivity and Paths: Cuts and Connectivity, k-connected Graphs, Network Flow Problems

Coloring and Planarity: Vector Colorings, k-chromatic graphs, Euler's Formula, Chacterization of Planar Graphs, Parameters of Planarity

Edges and Cycles: Line-Graphs and Edge Coloring, Hamiltonian Cycles, Planarity, Coloring and Cycles

Text books:

1. West, Douglas Brent. Introduction to graph theory. Vol. 2. Upper Saddle River: Prentice hall, 2001.

Reference books:

1. Martin Charles Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, 1980.

2. Bondy, John Adrian, and Uppaluri Siva Ramachandra Murty. Graph theory with applications. Vol. 290. London: Macmillan, 1976.

3. Tutte, William Thomas, and William Thomas Tutte. Graph theory. Vol. 21. Cambridge university press, 2001.

4. William Kocay and Donald L. Kreher, Graphs, Algorithms, and Optimization, CRC Press, 2005.

5. https://onlinecourses.nptel.ac.in/noc22ma10/preview

CS432	**Mobile Computing** **3-0-3** **Elective**
Pre-requisite	Computer Networks Effective from: 12/4/2023
Course Objectives	1. To learn basics of wireless transmission basics and medium access protocols. 2. To learn basic Wi-Fi technology and its protocols for working with any future wireless networks. 3. To learn and understand cellular technologies such as 2G, 2.5G, 3G and 4G. 4. To learn mobile IP, mobile TCP, wireless ad-hoc routing algorithms. 5. To learn WAP for implementing mobile applications.
Course Outcomes	At the end of the course, student will be able to: 1.Understanding the basic Wi-Fi and cellular technologies. 2. Choosing and applying right protocol messages for DCF, PCF, need of RTS/CTS addressing wireless network issues. 3. Evaluating cellular protocols such as 2G, 3G, and 4G in basic use cases. 4. Analyzing mobile IP, mobile TCP and WAP of mobile applications. 5. Designing and implementing ns-3 experiments related to Wi-Fi and cellular networks.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1		2									1		
CO2		1	2						1					
CO3			2	2	2		1					1		1
CO4			1	2	1									
CO5		2	2	2	3				1			1		1

Detailed Syllabus

Unit-1: Introduction to wireless transmissions: signals, frequency, channels, propagation losses, and medium access protocols: SDMA, TDMA, FDMA, OFDM, OFDMA, CDMA

Unit-2: Wi-Fi technologies and protocols: introduction to Wi-Fi protocols (802.11) CSMA/CD, Wi-Fi LANs operating modes (DCF and PCF), RTS/CTS, power saving modes, QoS, ns-3 simulations

Unit-3: Cellular technologies and protocols: Introduction to 2G, 3G and 4G technologies, architectures, GSM, UMTS and LTE protocols, ns-3 simulations

Unit-4: Mobile network and transport protocols: Need of Mobile IP, Mobile IP implementation, DHCP, ad-hoc network routing algorithms, Need of mobile TCP, comparison of TCP vs mobile TCP, mobile TCP features, ns-3 simulations

Unit-5: Mobile Applications Protocols : wireless application architectures, protocols stacks Wireless datagram protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment, example applications

Text books:

1. MOBILE COMMUNICATIONS by JOCHEN H SCHILLER (Author) Research papers and Internet sources

References:

1. Advanced Network Simulations Simplified - Dr Anil Kumar Rangisetti (Author), Packt

Other sources:

1. NS-3: https://www.nsnam.org/

CS433	**Pattern Recognition** **3-0-3** **Elective**
Pre-requisite	Programming, linear algebra, Probability and Statistics Effective from: 12/4/2023
Course Objectives	1. To motivate the students in the importance of pattern analysis and classification. 2. To introduce the extraction of pattern from unstructured data like text, speech, images, etc. 3. To expertise the students in mathematical formulation of problems through pattern vectors. 4. To exercise solving of non-linear problems through modeling in pattern space and differentiate amongst different machine learning paradigms. 5. To develop programming skills in developing pattern recognition applications.
Course Outcomes	1. Find patterns from structured and unstructured data and interpret in form of dataset. 2. Identify meaningful information and features from patterns extracted out of a given dataset 3. Inspect solutions to determine the relationship among the patterns through different machine learning paradigms 4. Predict and evaluate decisions for some unseen data based on analysis of the available patterns 5. Choose appropriate algorithms in solving linearly and non-linearly separable pattern classification problems

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**
CO1	**1**					1			2				**3**	**1**
CO2	**2**	**1**	**1**	**3**	**1**								**2**	**3**
CO3	2	**2**	**2**	**3**	**1**								**3**	**2**
CO4	**1**	2	**3**	**3**	1								**3**	**2**
CO5	3		3	**2**	2		**1**					**1**	2	**3**

Detailed Syllabus

Introduction to Patterns, machine intelligence, features, dimensionality, data visualization, correlation, learning paradigms, evaluation of performances, Introduction to data processing using Python
Basic decision making: Naïve Bayesian theory, KNN, K-means clustering, basics of kernel functions, distance measures, regression, classification, separability of data, data standardization, Introduction to Python Programming, decision making in real-life, curse of dimensionality, Cover's theorem, SVM, radial basis functions and solution to the separability problem, parametric vs non-parametric learning
Introduction to ANN, history computation vs time, intelligence in machines, decision making by ANN, weight updation, gradient descent algorithm, cost function, activation function, Pattern classification using self-organising feature maps, problems in ANN, vanishing and exploding gradient problems
CNN, working with CNN for image recognition, auto-encoders, designing of movie recommendation system using auto-encoder, RNNS, LSTMs, temporal sequencing in stock price monitoring Auto-regressive models: GANs, deep fake implementation with GANs, RBM, Deep belief networks and applications

Text books:

1. Duda, Richard O., and Peter E. Hart. Pattern classification. John Wiley Sons, 2006.

Reference books:

1. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.
2. Haykin, Simon. Neural networks and learning machines, 3/E. Pearson Education India, 2010.

CS434	Cloud Computing 3-0-3 Elective
Pre-requisite	Operating systems and computer networking Effective from: 12/4/2023
Course Objectives	1. To introduce cloud service models, deployment models, and infrastructures. 2. To learn various virtualization technologies suitable for cloud computing environments. 3. To explore the importance of lxc and docker technologies. 4. To learn the importance of programming networks and virtualization of network services. 5. To explore various details of giant cloud service providers: Amazon cloud, Google cloud services, etc.
Course Outcomes	At the end of the course, student will be able to: 1. Demonstrate about variety of cloud computing service and deployment models. 2. Identify right cloud service and deployment models for various use cases. 3. Build sample cloud environments, and deploying application using Docker and virtual machines. 4. Learn how to use programmable network and cloud environments for addressing flexibility, reliability and scalability issues. 5. Study case studies of large scale file systems, and database applications design and implementation

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	1		1				
CO2		2	1	1			
CO3	2		3		3	2	
CO4	2		3	2	3	2	
CO5	1	2				2	
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2		2					
CO3				3			
CO4				2			
CO5		3		3			2

Detailed Syllabus

Unit-1 Introduction to Cloud computing: Cloud computing definition, service models: Software as a Service (SaaS), features of SaaS and benefits, Platform as a Service (PaaS), features of PaaS and benefits, Infrastructure as a Service (IaaS), features of IaaS and benefits, FaaS, Service providers, challenges and risks in cloud adoption. Cloud deployment model: Public cloud, Private clouds, Community clouds, Hybrid clouds - Advantages of Cloud computing.

Unit-2 Virtualization and cloud computing: Need of virtualization. cost, administration, fast deployment, reduce infrastructure cost, limitations Types of hardware virtualization: Full virtualization, partial virtualization, para virtualization Desktop virtualization: Software virtualization, Memory virtualization, Storage virtualization, Data virtualization, Network virtualization, Hypervisors and Virtual machines

Unit-3 Cloud set up technologies: Quick introduction to LXCs, name spaces, cgroups, Introduction to Docker architecture, docker images, container, services, storage and networking and docker swarm architecture

Unit-4 Cloud networking technologies: Virtual networks, VLANs, Overlay networks, Importance of programmable networks and their architecture: SDN architecture, OpenFlow protocol, OpenFlow switches, Flow tables, Flow rules, SDN controller, introduction to NFV.

Unit-5 Cloud Case studies: Amazon AWS, Google file system, hadoop, big table, HBase.

Text books:

1. Cloud Computing: Concepts, Technology Architecture, 2014 by Erl (Author)

2. R. Buyya, C. Vecchiola and S. T. Selvi, Mastering Cloud Computing Foundations and Applications Programming, Morgan Kaufmann, Elsevier, 2013.

References:

1. Research papers and Internet sources related to cloud computing and technologies.

Other sources:

1.https://www.docker.com/

2. https://linuxcontainers.org/

3. http://mininet.org/

CS435	**Parallel and Dsitributed Computing**	**3-0-3**	**Elective**
Pre-requisite	Students must be able to have a good understanding of Computer Networks, Operating Systems, and Computer Architecture.		Effective from: 12/4/2023
Course Objectives	1. To understand the basic concepts of parallel and distributed computing. 2. To learn about the different models of parallel and distributed systems. 3. To understand the techniques for designing and implementing parallel and distributed algorithms. 4. To gain practical experience in parallel and distributed programming. 5. To learn about the challenges and limitations of parallel and distributed computing.		
Course Outcomes	At the end of the course, student will be able to: 1. Understand the basic concepts of parallel and distributed computing and identify the differences between them. 2. Analyze and compare different models of parallel and distributed systems. 3. Design and implement parallel and distributed algorithms using different programming languages and tools. 4. Evaluate the limitations and challenges of parallel and distributed computing and propose possible solutions. 5. Revise a single thread program to a parallel program and vice-versa.		

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1		2	2								3	2	1
CO2	2	3	2	3	2								2	2
CO3	3	2	3	2	3								1	1
CO4	1	2	1	3	1								1	1
CO5	1	3	3	3	3								2	2

Detailed Syllabus

Introduction: Why Parallel Computers?, Parallel vs Serial Execution, Shared vs Distributed Memory, Methodical Design of Parallel Programs, Measuring Performance of Parallel Programs, Demand for Computational Speed, Types of Parallel Computers.

Message-Passing Computing: Architectural Features of Message-Passing Multicomputers, Potential for Increased Computational Speed, Basics of Message-Passing Programming, Evaluating Parallel Programs, Debugging.

Partitioning and Divide and Conquer Strategies: Partitioning, Divide- and Conquer, Pipeline Technique, Computing Platform for Pipelined Applications, Pipeline Program Examples.

Synchronous Computations: Synchronization, Computations, Iteration Program Examples.

Load Balancing and Termination Detection: Load Balancing, Dynamic Load Balancing, Distribution Termination Detection.

Text books:

1. B. Wilkinson and M. Allen. Parallel Programming Techniques Applications Using Networked Workstations Parallel Computers, 2nd ed. Toronto, Canada: Pearson, 2004.

CS446	**Marketing Science and Predictive Analysis**	**3-0-3**	**Elective**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To understand the importance of marketing science and its role in driving business decisions 2. To learn about the different methods and techniques used in marketing science and predictive analysis 3. To cultivate critical thinking and problem-solving skills related to marketing challenges 4. To develop an understanding of how to apply marketing science and predictive analysis methods to real-world business scenarios		
Course Outcomes	1. Students will be able to identify different methods and techniques used in marketing science and predictive analysis. 2. Students will be able to evaluate and select the most appropriate methods and tools for a given business problem. 2. Students will be able to apply statistical models and algorithms to real-world scenarios and generate insights and recommendations for business actions. 4. Students will be able to communicate complex concepts and analysis results with clarity and persuasiveness.		

Detailed Syllabus

Introduction to marketing science and predictive analysis, Explanation of the role of marketing science in business decision-making

Overview of different kinds of data analysis - descriptive, diagnostic, predictive and prescriptive, Career opportunities for marketing science professionals

Data collection and processing, Types of data - qualitative and quantitative, Sampling methods and their importance, Data cleaning, transformation and preparation techniques

Introduction to data visualization tools and techniques, Statistical models and algorithms, Regression analysis, its different types, and applications in marketing science, Cluster analysis, segmentation analysis, and factor analysis, Time series analysis and forecasting, its importance, and applications in marketing science

Machine learning, Introduction to supervised and unsupervised learning, Types of algorithms used in machine learning - decision trees, random forests, artificial neural networks, etc, Applications of machine learning techniques in marketing science, Advanced analytics techniques, A/B testing for experimentation, Social media analytics and web analytics

Text mining and sentiment analysis, Real-world case studies and problem-solving exercises, Evaluation and selection of methods and tools for marketing problems, Generating insights and recommendations from analysis results, Communicating analysis findings with clarity and persuasiveness.

Reference books:

1. "Marketing Analytics: Data-Driven Techniques with Microsoft Excel" by Wayne L. Winston.
2. "Data-Driven Marketing: The 15 Metrics Everyone in Marketing Should Know" by Mark Jeffery.
3. "Marketing Metrics: The Definitive Guide to Measuring Marketing Performance" by Paul W. Farris, Neil T. Bendle, Phillip E. Pfeifer, and David J. Reibstein.

CS450	**Data Analysis and Visualization** **3-0-3** **Elective**
Pre-requisite	Nil Effective from: 12/4/2023
Course Objectives	1. Understand importance of analytics and visualization in data-rich era 2. Review probability, statistics and random processes 3. Learn estimation theory basics 4. Study machine learning techniques: clustering, regression, classification 5. Master data visualization principles, VTK for 3D graphics and deep learning
Course Outcomes	The students should be able to: 1. Choose suitable software and chart formats for modeling data 2. Utilize data analysis tools within the numpy and pandas library. 3. Effectively load, clean, transform, merge, and reshape data. 4. Apply statistical analysis and regression techniques using R. 5. Apply acquired knowledge to solve real-world data analysis problems.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1	2	3	1	1	-	2	1	2	1	3	1
CO2	1	1	1	1	3	1	-	-	2	-	1	1	3	1
CO3	1	2	1	1	3	1	-	1	2	1	1	1	3	1
CO4	2	3	3	3	3	3	1	-	2	-	2	1	3	1
CO5	1	1	1	2	2	2	1	-	1	1	2	3	3	1

Detailed Syllabus

Data and information, Data analysis and data analytics, why data analysis?, Data analysis process; Data Analysis Methods- Qualitative and Quantitative Data Analysis.

Excel for data analysis- Essential data analysis functions in excel; Data Cleaning and Transformation-Removing duplicates, Handling missing values, Text-to-columns for data separation, Using conditional formatting for data validation; Basic Formulas and Functions (sum, avg, min, max, if, etc); Advanced functions (sumifs, countifs, concatenate etc); Data Visualization-Creating basic charts (bar, line, pie), Pivot Tables and charts, Using the Analysis ToolPak-Descriptive statistics, Case studies.

Statistics: Introduction to statistics, Descriptive statistics-Central tendency: Mean, Median, Mode; Variance and Standard Deviation, Median Absolute Deviation, Skewness and Kurtosis, Coefficient of variation (CV), Covariance and correlation (Pearson Correlation, Spearman Correlation), Quantiles- Quartiles, deciles and Percentiles, Five number summary: Q1, Q3, Range & IQR.

Distributions: Distributions: Frequency distribution, Probability Density Function, Cumulative frequency distribution, CDF, Uniform distribution, Normal distribution with its PDF and CDF, Population and sample, Central limit theorem, Standardization of data, Checking for normal distribution - QQ plot, Conversion to normal distribution- log, reciprocal, sqrt, Box cox transformation, yoejohnson.

Python for Data Analysis; Introduction of iPython and Jupyter Notebook, Introduction to Numpy Libraries: NumPy Arrays in python, Arrays and vectorized computation-Array operations-indexing, slicing, broadcasting, Statistical functions, advanced NumPy features.

Getting Started with Pandas: Introduction to pandas Data Structures, Essential Functionality, Summarizing and Computing Descriptive Statistics. Data Loading, Storage and File Formats. Reading and Writing Data in Text Format, Data Cleaning and Preprocessing: Handling missing data (isna, dropna, fillna), Removing duplicates, Renaming columns; Data Filtering and Selection- Boolean indexing, Using loc and iloc for selection, Querying DataFrames; Data Wrangling: Hierarchical Indexing, Combining and Merging Data Sets Reshaping and Pivoting.

Data Visualization: matplotlib: Basics of matplotlib, plotting with pandas and seaborn, other python visualization tools, Graphical Representation and Statistical plotting of data (scatter plots, line plot, bar charts, pie charts, box plots, density plots, histograms) Essential Python Libraries: NumPy, pandas, matplotlib, seaborn, SciPy, scikit-learn, statsmodels; Integrating Pandas with other libraries (NumPy, Matplotlib, Seaborn, etc.)

Data Analysis Case Studies: Applying Pandas to real-world datasets, Exploratory Data Analysis (EDA) using Pandas, Solving common data analysis problems with Pandas. (titanic, planets, bike sales, hotel tips datasets etc) R for data analysis- Overview of R Programming, Essentials of R - variables, operators, mathematical and logical operations, loops and repeat, Inputting data - data types, writing data, creating a vector and vector operation, initializing data frame, control structure.

Visualize graphs using R:Graphical Representation and Statistical plotting of data (scatter plots, line plot, bar charts, pie charts, box plots, histograms) color the plots, add text to plots. Understand the statistical concepts using R; Statistical modeling – hypothesis testing, maximum likelihood, regression, classification, curse of dimensionality using R.

Introduction to other data visualization tools: Tableau, Looker and Google Refine, Creating a Data Visualization Poster is a mandatory project for all students enrolled in this course.

Reference books:

1. McKinney, W.(2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython. 2nd edition. O'Reilly Media.

2. R For Data Analysis In Easy Steps by Mike McGrath, BPB Publications.

3. O'Neil, C., Schutt, R. (2013). Doing Data Science: Straight Talk from the Frontline O'Reilly Media

4.Few, Stephen. Now You See It: Simple Visualization Techniques for Quantitative Analysis

5. Datasets:https://data.gov.in/

Others:

6. https://towardsdatascience.com/the-power-of-visualization-in-data-science-1995d56e4208

7. https://hbr.org/2016/06/visualizations-that-really-work

8. https://rhandbook.wordpress.com/tag/slicing-and-dicing-of-data-frame/

CS460	Virtual Reality and Augmented Reality	3-0-3	Elective
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. Apply concepts of Virtual Reality 2. Apply concepts of Augmented Reality 3. Integrate sensors with AR/VR system 4. Design AR/VR application for a given task 5. Analyze existing AR/VR systems		
Course Outcomes	1. Apply concepts of Virtual Reality 2. Apply concepts of Augmented Reality 3. Integrate sensors with AR/VR system 4. Design AR/VR application for a given task 5. Analyze existing AR/VR systems		

Detailed Syllabus

Introduction to Virtual Reality, History, Overview of Various Realities, Immersion. Perception: Objective Subjective Reality, Perceptual Modalities, Perception of Space Time.
Content Creation: Environmental design, Affecting Behavior, VR Content Creation. Interaction: VR Interaction Concepts, Input Devices, Interaction Patterns Techniques. Iterative Design of VR: Philosophy of Iterative Design, The Define stage, Make stage and Learn stage.
Software Development in Virtual Reality. Introduction to Augmented Reality, Displays, Tracking, Computer Vision for Augmented Reality, Calibration Registration, Visualization, Interaction, Modeling Annotation, Authoring, Navigation, Collaboration.
Software Development in Augmented Reality: Applications of VR/AR in Entertainment, Medical, Manufacturing, Education, etc., Future of VR/AR.

Reference books:

1. Jason Jerald, "The VR Book: Human-Centered Design for Virtual Reality", ACM and Morgan Claypool Publishers, 2015.
2. Dieter Schmalstieg, Tobias Hollerer, "Augmented Reality, Principles and Practice", Addison Wesley (Pearson Education), 2016.
3. Jesse Glover and Jonathan Linowes, "Complete Virtual Reality and Augmented Reality Development with Unity", Packt Publishers, 2019.

CS462	Wireless Networked Systems 3-0-3 Elective
Pre-requisite	Computer Networks Effective from: 12/4/2023
Course Objectives	1. To learn basic Wi-Fi technology and its protocols for working with any future wireless networks. 2. To explore and learn a range of Wi-Fi technologies (802.11a/b/g) and features. 3. To explore and learn advanced Wi-Fi technologies (802.11n/ac) and high-efficiency wireless networks (802.11ax) supporting features. 4. To learn how to set up and evaluate a variety of wireless LANs and ad-hoc networks. 5. To learn basic and advanced Wi-Fi technologies by experimenting with simulators.
Course Outcomes	At the end of the course, student will be able to: 1. Exploring ns-3 for wired networking experiments. 2. Experiment with basic Wi-Fi technologies and wireless network 3. Analyzing wireless ad-hoc networks and routing algorithms. 4. Evaluating 802.11n, 802.11ac and 802.11ax networks and features. 5. Set up and evaluate advanced wireless network setups.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1	1	2	2	3	3		
CO2	2		1	2	2		
CO3	2		1	2			
CO4	2		1	2			
CO5	1		2	2	2		
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4		1					1
CO5		1			1		1

Detailed Syllabus

Unit-1: Quick Introduction to NS-3 and its features, NS-3 Key building blocks, applications, explore and evaluate NS-3 ethernet modules: CSMA/P2P.

Unit-2: Basics of Wi-Fi, 802.11a,b,g technologies, Wireless LANs, DCF, PCF, RTS/CTS modes, fragmentation mode, power saving modes, operating modes, and QoS feature, basic simulations using ns-3 supporting Wi-Fi features

Unit-3: Importance of wireless ad-hoc networks, protocols, routing algorithms, challenges in deployment, wireless ad-hoc networks evaluations using ns-3,

Unit-4: 802.11n, 802.11ac: Physical layer (Channel bonding, MIMO, higher MCS) and MAC layer enhancements (frame aggregations, scheduling), backward compatibility, standards specific deployment modes, and ns-3 supporting 802.11n/ac features

Unit-5: Hi-Efficiency wireless networks (Wi-Fi 6): 802.11ax, radio resources management, scheduling, interference handling and ns-3 supporting Wi-Fi 6 Simulations

Text books:

1.MOBILE COMMUNICATIONS by JOCHEN H SCHILLER (Author)

2. Wi-Fi 6 6E For Dummies®, Extreme Networks Special Edition. Published by. John Wiley Sons

References:

1. Advanced Network Simulations Simplified - Dr Anil Kumar Rangisetti (Author), Packt
2. Research papers and Internet sources related to cloud computing and technologies.

Other sources:

1. NS-3: https://www.nsnam.org/

CS463	**Post Quantum Cryptography** **3-0-3** **Elective**
Pre-requisite	Cryptography, Linear Algebra, Probability Theory, Design and Analysis of Algorithm Effective from: 12/4/2023
Course Objectives	1. Understanding Cryptographic Techniques: Gain proficiency in symmetric key encryption, public key encryption, digital signatures, and hash functions. 2. Investigating Quantum Algorithms: Explore Shor's algorithm for factoring and Grover's algorithm for searching, emphasizing their implications for cryptography. 3. Exploring Lattice-Based Cryptography: Study lattice-based cryptography, focusing on practical protocols like Kyber (Key Encapsulation Mechanism) and Dilithium (Digital Signatures). 4. Understanding Code-Based Cryptography: Grasp the fundamentals of code-based cryptography and explore classical encapsulation schemes such as McEliece and HQC. 5. Gaining Insights into Post-Quantum Cryptosystems: Gain insight into multivariate-based cryptography and develop a basic understanding of isogeny-based cryptography.
Course Outcomes	At the end of the course, the student will be able to: 1. Apply Cryptographic Techniques: Apply cryptographic techniques to encrypt messages securely and sign messages for authentication. 2. Evaluate Quantum Algorithms: Assess the impact of quantum algorithms like Shor's and Grover's on cryptographic protocols. 3. Implement Lattice-Based Cryptographic Schemes: Implement and analyze lattice-based cryptographic schemes like Kyber and Dilithium, focusing on their resistance to quantum attacks. 4. Implement Code-Based Cryptography: Implement code-based cryptography schemes such as McEliece and HQC, understanding their resilience against quantum computing threats. 5. Understand Advanced Cryptographic Principles: Understand the underlying principles of multivariate-based and isogeny-based cryptography, preparing for future developments in cryptographic protocols.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO1							
CO2							
CO3							
CO4							
CO5							
CO	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1							
CO2							
CO3							
CO4							
CO5							

Detailed Syllabus

Module - I: Introduction to Cryptography

Symmetric Key Encryption – AES Block Cipher, Public Key Encryption – RSA Encryption scheme, El Gamal Encryption scheme. Key Encapsulation Scheme. Cryptographic Hash Function and its security properties. Digital Signature Scheme – Concrete Signature protocols: RSA, ECDSA, Schnorr.

Module - II: Quantum

Grover's algorithm, Its impact on Symmetric key encryption, Hash function. Shor's algorithm and its impact on Factorization and Discrete Logarithm Problems.

Module - III: Lattice-based Post-Quantum Cryptosystems:

Overview of lattice-based cryptography: Kyber and Saber (NIST post-quantum candidates). Classical constructions - NTRU, NTRU Prime. Digital Signature Algorithm: Dilithium, Falcon.

Module - IV: Multivariate-based Post-Quantum Cryptosystems:

Overview of Multivariate-based cryptography. Unbalanced oil and vinegar scheme, Rainbow signature scheme, Hidden Field Equations scheme.

Module - V: Code-based and Isogeny-based Post-Quantum Cryptosystems:

Code-based Cryptosystems: Overview of Code-based cryptography, Classic McEliece encapsulation scheme. Code-based signatures scheme.

Isogeny-based Cryptosystems: Overview of Isogeny-based cryptography, Supersigular Isogeny-based Key Exchange (SIKE), Digital Signature Algorithm based on Isogeny.

Text books:

1. Daniel J. Bernstein, Johannes Buchmann and Erik Dahmen, "Post-Quantum Cryptography", Springer, 2009.

Useful Links

1. https://pqcrypto.org/index.html
2. https://csrc.nist.gov/projects/post-quantum-cryptography

CS465	Web Technologies 3-0-3 Elective
Pre-requisite	Basic Programming Skills Effective from: 12/4/2023
Course Objectives	1. To be familiar with client server architecture. 2. To understand client and server-side scripting and their applicability. 3. To gain the skills needed for entry into web application and development careers. 4. To design and develop web applications. 5. To understand different architectures and predict the apt architecture for a given problem.
Course Outcomes	At the end of the course, student will be able to: 1. Select a suitable framework for the website based on the necessity. 2. Compare different languages for integrating media and user interaction in both front end and back end elements of a Web site. 3. Experiment and analyse the web data using web analytics. 4. Assess the effectiveness of a website through different testing and debugging techniques. 5. Adapt and re-build the present-day websites based on the issues that affect them.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1		2	2	1	2								2	2
CO2		1	1	2	2									
CO3		2	2	2									1	1
CO4	1	2	1	2	3								2	
CO5	1	1	2	3								3	2	

Detailed Syllabus

Introduction: Introduction to Computers, Web browser basics, Web 2.0, and Web 3.0

The Ajax Client: Introduction to XHTML, CSS, Javascript, XML, RSS, DOM, Ajax-enabled Rich Internet Applications

Rich Internet Client Applications: Adobe Flash CS3, Adobe Dreamviewer

Rich Internet Server Applications: IIS, Apache, SQL, MY SQL, Java DB, PHP, ASP.NET, Ruby, ASP.NET Ajax

Web Services and Web Applications: Java Server, Web Services, Ajax Enabled Web Services

Text books:

1. Deitel, Deitel and Nieto, Internet and Worldwide Web - How to Program, 4th Edition, PHI, 2008.

Reference books:

1. Bai and Ekedhi, The Web Warrior Guide to Web Programming, 3rd Edition, Thomson, 2008.

CS466	**Cryptography and Network Security** **3-0-3** **Elective**
Pre-requisite	Discrete Mathematics, Probability Theory, Theory of Computation or Automata Theory Effective from: 12/4/2023
Course Objectives	1. Mathematical maturity to understand the cryptosystem and its cryptanalysis. 2. Understand previous attacks on the classical cryptosystems, utilize the lesson learned to prevent future attacks. 3. Analyze the symmetric key encryption and hashing algorithms. 4. Analyze the public key encryption and signature algorithms. 5. Apply cryptographic algorithms to build secure protocols. 6. Design of secure protocols to solve real world scenarios.
Course Outcomes	At the end of the course, student will be able to: 1. Understand the cryptosystem and its cryptanalysis. 2. Understand the impact of the cryptanalysis on the classical cryptosystems, utilize the lesson learned to prevent future attacks. 3. Understand concept of symmetric key encryption, public key encryption, hashing and signature algorithms. 4. Understand the application of cryptographic algorithms to build secure protocols. 5. Solve real world scenarios by designing secure protocols.

CO-PO/PSO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	1											
CO2	3	2	1											
CO3	2	1	3											
CO4		1	3	2										
CO5			1	2		3								
CO6				1	2	3								

Detailed Syllabus

Module - I

Course Overview, Mathematical background: Modular Inverse, Extended Euclid Algorithm, Fermat's Little Theorem, Euler Phi-Function, Euler's theorem. Symmetric-key Encryption, Historical Ciphers, Stream Ciphers. Provably secure Instantiation of PRG, Practical Instantiation of PRG, CPA-Secure Ciphers from PRF, Modes of Operations of Block Ciphers, DES, AES. Message Authentication Codes (MAC), Information-theoretic Secure MAC, Cryptographic Hash Functions. Birthday Attacks on Cryptographic Hash Functions.

Module - II

Public Key Encryption: Discrete-Logarithm Problem, Computational Diffie-Hellman Problem, Decisional Diffie-Hellman Problem, Elliptic-Curve Based Cryptography and Public-Key Encryption, El Gamal Encryption Scheme, RSA Assumption, CCA - secure Public-key Hybrid Ciphers Based on Diffie-Hellman Problems and RSA-assumption.

Module - III

Digital Signatures: Schnorr Signature, Digital-signature Standard Algorithm (DSA), Elliptic Curve DSA. Certificate Management, Public Key Infrastructure (PKI). Zero Knowledge Proof System, ZK for Graph Isomorphism.

Module - IV

Introduction to network security: Network security threats; Vulnerabilities - Denial-of-service / Distributed denial-of-service attacks; Spoofing, Man-in-the-middle, Replay, TCP/Hijacking, Fragmentation attacks, Weak keys, Mathematical attacks, Port scanning, Dumpster diving, Birthday attacks, Password guessing.

Module - V

Security Protocols: Network and transport layer security - SSL/TLS, IPsec IKE; IPsec AH, ESP; Application security - Kerberos; S/MIME; PGP.

Text books:

1. Douglas R. Stinson and Maura Paterson, Cryptography: Theory and Practice, Chapman and Hall/CRC publication, Standard Edition, 2018.

2. Behrouz A. Forouzan, Introduction to Cryptography and Network Security, Mc Graw Hill Publication, 2020.

Reference books:

1. Jonathan Katz and Yehuda Lindell, Introduction to Modern Cryptography, Chapman and Hall/CRC publication, 2nd edition, 2014.

2. A. Menezes, P. Van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 2004.

3. Jean-Philippe Aumasson, Serious Cryptography - A Practical Introduction to Modern Encryption, No Starch Press Publication, 2017.

Online Courses:

1. NPTEL Course on Foundations of Cryptography By Prof. Ashish Choudhury, https://onlinecourses.nptel.ac.in/noc22_cs03/preview

2. NPTEL Course on Cryptography and Network Security By Prof. Sourav Mukhopadhyay, https://onlinecourses.nptel.ac.in/noc21_cs16/preview

3. Coursera on Cryptography I by Prof Don Boneh, https://www.coursera.org/learn/crypto.

CS468	**Remote Sensing and GIS 3-0-3 Elective**
Pre-requisite	Basic programming knowledge Effective from: 12/4/2023
Course Outcomes	1. Outline various terminologies related to remote sensing and processing geographical information obtained from different types of satellites, aerial vehicles, LiDAR, Radar sensors 2. Understand the basic phenomenon of capturing data in the electromagnetic spectrum and the corrections required for processing raw data from remote sensing instruments 3. Visualisation of multiresolution and multimodal data effectively through tools and image-band manipulation techniques 4. Discovery of knowledge from remotely sensed data for effective decision-making, denoising, compression, and analysis 5. Assess the state-of-the-art technologies regarding the application of remote-sensing data to solve real-world problems related to life sustainability

Remote sensing and GIS (CS468)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2
CO1	2													1
CO2	2		1	2									1	
CO3		3	2	2	3								2	3
CO4	2	1	2	2									2	2
CO5		2	3	3			3					1	2	3

Detailed Syllabus

Introduction: Definitions with motivating examples of remote sensing and geographical information processing, applications of remote sensing technologies in connection to geographical information processing, history and scope of remote sensing, electromagnetic spectrum, active versus passive remote sensing, data capturing through Earth observation satellites, multispectral versus hyperspectral remote sensing, aerial photogrammetry, Microwave, LiDAR and thermal remote sensing, refraction, reflection, scattering, absorption, radiometric, geometric and atmospheric corrections, microwave and infrared remote sensing
Geovisualisation: Satellite footprint, swath area, views, spatial, spectral and temporal resolutions, greyscale display, true-colour display, false-colour display, pan-sharpening of images, visualisation through spectral bands, vegetation indices, satellite image enhancement, visualisation of multispectral and hyperspectral images, interpreting SAR images, obtaining land images from RGB and near-infrared bands, obtaining geographical data from satellite images, digital cartography, tools for geovisualisation
Knowledge discovery: Types of spatial and temporal data, data mining from spatiotemporal data, feature analysis, feature selection, data compression, denoising, classification, summarisation of satellite data, clustering, change detection, open research problems
Applications domains: interpretation of bands, agriculture, forestry, coastal surveillance, leaf spectral analysis, bathymetry, phenotyping, plant pathology, ice monitoring, land-cover and land-use mapping, habitation growth, urban surveillance

Reference books:

1.Campbell, James B., and Randolph H. Wynne. Introduction to remote sensing. Guilford press, 2011.

2. Lillesand, T.M., Kiefer, R.W. and Chapman, J.W., “Remote Sensing and Image Interpretation”, (5th Ed.), John Wiley Sons, 2007.

Others:

1. https://ecampusontario.pressbooks.pub/remotesensing/front-matter/welcome-to-remote-sensing/

2. Laurini, Robert. "A conceptual framework for geographic knowledge engineering." Journal of Visual Languages Computing 25.1 (2014): 2-19.

3. https://nptel.ac.in/courses/105108077

CS469	**Gen AI**	**3-0-3**	**Open Elective**
Pre-requisite	Nil		Effective from: 12/4/2023
Course Objectives	1. To understand the fundamental principles and theories underlying General Artificial Intelligence (Gen AI). 2. To explore the various architectures, algorithms, and techniques used in Gen AI systems and to analyze the ethical, societal, and economic implications of Gen AI. 3. To develop practical skills in designing, implementing, and evaluating Gen AI applications. 4. To foster the ability to critically assess the potential and limitations of Gen AI technologies.		
Course Outcomes	1. Demonstrate a thorough understanding of the core concepts and methodologies of General Artificial Intelligence. 2. Apply advanced algorithms and techniques to solve complex problems in Gen AI. 3. Critically evaluate the ethical and societal impacts of Gen AI technologies. 4. Design and implement Gen AI applications using appropriate tools and frameworks. 5. Analyze and discuss current research trends and future directions in General Artificial Intelligence.		

Detailed Syllabus

Introduction to General AI, History and Evolution of AI, Foundations of General AI, AI vs. Narrow AI, Machine Learning Basics, Deep Learning Fundamentals, Neural Networks, Advanced Neural Network Architectures, Reinforcement Learning

Unsupervised Learning Techniques, Supervised Learning Techniques, Natural Language Processing, Computer Vision, Robotics and AI, Cognitive Computing, AI in Decision Making, AI in Problem Solving, Knowledge Representation, Reasoning and Inference

AI in Autonomous Systems, Ethical Issues in AI, Societal Impact of AI, AI and Employment, AI in Healthcare, AI in Finance, AI in Education, AI in Security, AI and Privacy, AI Tools and Frameworks, Future Directions in General AI, AI Research Methodologies, Evaluation Metrics for AI, AI in Real-World Applications

Reference books:

1. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
2. "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville
3. "The Quest for Artificial Intelligence: A History of Ideas and Achievements" by Nils J. Nilsson