RCOEM

Shri Ramdeobaba College of Engineering and Management, Nagpur

SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT

An Autonomous College affiliated to Rashtrasant Tukadoji Maharaj Nagpur University,
NAGPUR – 440013
Nagpur,Maharashtra(INDIA)

PROGRAMME SCHEME

2023-2024

B.TECH. (ELECTRONICS AND COMPUTER SCIENCE)

SEMESTER-I											
Sr. No.	Category	Course Code	Course Title	Hours per week			Credits	Maximum Marks			ESE Duration(Hrs)
				L	T	P		Continuous Evaluation	End Sem Exam	Total	
1	BSC	PHT1007	Physics	2	1	0	3	50	50	100	3
2	BSC	PHP1007	Physics Lab	0	0	2	1	50	-	50	-
3	BSC	MAT1001	Applied Mathematics–I	2	1	0	3	50	50	100	3
4	BSC	MAP1001	Computational Mathematics Lab	0	0	2	1	50	-	50	-
5	BSC	CHP1002	Environmental Science Lab	0	0	2	1	50	-	50	-
6	ESC	ECST1001	Programming for Problem Solving	3	0	0	3	50	50	100	3
7	ESC	ECSP1001	Programming for Problem Solving Lab	0	0	2	1	50	-	50	-
8	VSEC	ECST1002	Fundamentals of Computer Engineering	2	0	0	2	50	50	100	2
9	VSEC	ECST1003	Electronics and Computer Workshop	1	0	0	1	50	-	50	-
10	VSEC	ECSP1003	Electronics and Computer Work shop Lab	0	0	2	1	50	-	50	-
11	HSSM-AEC	HUT1002	English for Professional Communication	2	0	0	2	50	50	100	2
12	HSSM-AEC	HUP1002	English for Professional Communication Lab	0	0	2	1	50	-	50	-
1	CCA	HUP0001-1	Liberal/	0	0	2	1	5	-	50	-

3		toHUP0001-10PEP0001-21PEP0001-22CHP0001-31 CHP0001-32	Performing Arts					0			
14	HSSM-VEC	HUT1004	Foundation course In Universal Human Value	1	0	0	1	50	-	50	-
TOTAL				**13**	**2**	**14**	**22**				
				29 Hrs.							

SEMESTER-II											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	BSC	CHT2006	Chemistry of Smart Materials	2	0	0	2	50	50	100	2
2	BSC	CHP2006	Chemistry of Smart Materials Lab	0	0	2	1	50	-	50	-
3	BSC	MAT2001	Applied Mathematics –II	2	1	0	3	50	50	100	3
4	ESC	ECST2001	Elements of IoT	3	0	0	3	50	50	100	3
5	ESC	ECSP2001	Elements of IoT Lab	0	0	2	1	50	-	50	-
6	ESC	ECST2002	Object Oriented Programming	2	1	0	3	50	50	100	3
7	ESC	ECSP2002	Object Oriented Programming Lab	0	0	2	1	50	-	50	-
8	PCC	ECST2003	Digital Electronics	3	0	0	3	50	50	100	3
9	PCC	ECSP2003	Digital Electronics Lab	0	0	2	1	50	-	50	-
10	HSSM-IKS	HUT2001	Foundational Literature of Indian Civilization	2	0	0	2	50	50	100	2
11	CCA	PET2001	Sports-Yoga-Recreation	1	0	0	1	50	-	50	
12	CCA	PEP2001	Sports-Yoga-Recreation Lab	0	0	2	1	50	-	50	-
TOTAL				**15**	**2**	**10**	**22**				
				27 Hrs.							

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

SEMESTER-III											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	ECST3001	Data Structures	2	1	0	3	50	50	100	3
2	PCC	ECSP3001	Data Structures Lab	0	0	2	1	50	-	50	-
3	PCC	ECST3002	Electronic Devices And Circuits	2	1	0	3	50	50	100	3
4	PCC	ECSP3002	Electronic Devices And Circuits Lab	0	0	2	1	50	-	50	-
5	PCC	ECST3003	Digital System Design	2	1	0	3	50	50	100	3
6	PCC	ECSP3003	Digital System Design Lab	0	0	2	1	50	-	50	-
7	MDM	ECST3004	Discrete Mathematics	2	0	0	2	50	50	100	2
8	OE		Open Elective-I/ MOOCs	2	0	0	2	50	50	100	2
9	HSSM	HUT3003	Managerial Economics	2	0	0	2	50	50	100	2
10	HSSM-VEC	ECST3005	Cyber Laws and Ethics in IT	2	0	0	2	50	50	100	2
TOTAL				**14**	**3**	**6**	**20**				
				23Hrs.							

SEMESTER-IV											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	ECST4001	Computer Architecture And Organization	2	1	0	3	50	50	100	3
2	PCC	ECSP4001	Computer Architecture And Organization Lab	0	0	2	1	50	-	50	-
3	PCC	ECST4002	Design and Analysis of Algorithms	2	1	0	3	50	50	100	3
4	PCC	ECST4003	Software Engineering	2	1	0	3	50	50	100	3
5	PCC	ECSP4003	Software Engineering Lab	0	0	2	1	50	-	50	-
6	MDM	ECST4004	Statistics for Data Analytics	3	1	0	4	50	50	100	3
7	OE		Open Elective-II/MOOCs	3	0	0	3	50	50	100	3
8	VSEC	ECSP4005	Software Laboratory-I	0	0	4	2	50	-	50	-
9	HSSM	MBT	Business Management and Entrepreneurship	2	0	0	2	50	50	100	2
TOTAL				**14**	**4**	**8**	**22**				
				26 Hrs.							

Exit option: Award of UG Diploma with additional 8 credits			
Exit Course			
1	Application Development(Android)	Online/Offline Certification Course	8
2	Software Engineer(Developer)		8

3	PCB Designer		8

SEMESTER -V											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	ECST5001	Operating System	2	1	0	3	50	50	100	3
2	PCC	ECSP5001	Operating System Lab	0	0	2	1	50	-	50	-
3	PCC	ECST5002	Embedded System Design	2	1	0	3	50	50	100	3
4	PCC	ECSP5002	Embedded System Design Lab	0	0	2	1	50	-	50	-
5	PCC	ECST5003	Digital VLSI Design	2	1	0	3	50	50	100	3
6	PCC	ECSP5003	Digital VLSI Design Lab	0	0	2	1	50	-	50	-
7	MDM	ECST5004	Machine learning	3	0	0	3	50	50	100	3
8	MDM	ECSP5004	Machine learning Lab	0	0	2	1	50	-	50	-
9	PEC	ECST5005	Programme Elective-I	3	0	0	3	50	50	100	3
10	PEC	ECSP5005	Programme Elective-ILab	0	0	2	1	50	-	50	-
11	OE		Open elective-III/ MOOCs	3	0	0	3	50	50	100	3
TOTAL				**15**	**3**	**10**	**23**				
				28 Hrs.							

SEMESTER-VI											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	ECST6001	Database Management System	3	0	0	3	50	50	100	3
2	PCC	ECSP6001	Database Management System Lab	0	0	2	1	50	-	50	-
3	PCC	ECST6002	System Verilog for Verification	3	0	0	3	50	50	100	3
4	PCC	ECSP6002	System Verilog for Verification Lab	0	0	2	1	50	-	50	-
5	PEC	ECST6003	Programme Elective-II	3	0	0	3	50	50	100	3
6	PEC	ECSP6003	Programme Elective-II Lab	0	0	2	1	50	-	50	-
7	PEC	ECST6004	Programme Elective-III	3	0	0	3	50	50	100	3
8	PEC	ECSP6004	Programme Elective-III Lab	0	0	2	1	50	-	50	-
9	MDM	ECST6005	Data Handling and Visualization	2	0	0	2	50	50	100	2
10	VSEC	ECSP6006	Software Laboratory-II	0	0	4	2	50	-	50	
11	FP	ECSP6007	Project-1	0	0	4	2	50	50	100	-
TOTAL				**14**	**0**	**16**	**22**				
				30 Hrs.							

Exit option: Award ofUGDegreewithadditional8credits			
Exit Course			
1	TBI Internship		8

2	Centre for Micro system Internship	Online/offline Certification Course	8
3	Research Internship		8

SEMESTER-VII											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PCC	ECST7001	Computer Networks	3	0	0	3	50	50	100	3
2	PCC	ECSP7001	Computer Networks Lab	0	0	2	1	50	-	50	-
3	PCC	ECST7002	SOC Design	3	0	0	3	50	50	100	3
4	PCC	ECSP7002	SOC Design Lab	0	0	2	1	50	-	50	-
5	PCC	ECST7003	Information Security And Cryptography	3	0	0	3	50	50	100	3
6	MDM	ECST7004	Digital Signal Processing	3	0	0	3	50	50	100	3
7	PEC	ECST7005	Programme Elective-IV	3	0	0	3	50	50	100	3
8	PRJ	ECSP7006	Project-2	0	0	4	2	50	50	100	-
9	FP	ECSP7007	Internship Evaluation	0	0	2	0				
TOTAL				**15**	**0**	**10**	**19**				
				25 Hrs.							

SEMESTER-VIII											
Sr. No.	**Category**	**Course Code**	**Course Title**	**Hours per week**			**Credits**	**Maximum Marks**			**ESE Duration(Hrs)**
				L	**T**	**P**		**Continuous Evaluation**	**End Sem Exam**	**Total**	
1	PEC	ECST8001	Program Elective-V	3	0	0	3	50	50	100	3
2	PEC	ECST8002	Program Elective-VI	3	0	0	3	50	50	100	3
3	PRJ	ECSP8003	Project-3	0	0	12	6	50	50	100	-
TOTAL				**6**	**0**	**12**	**12**				
				18 Hrs.							
OR											
1	Internship/ OJT	ECSP8006	Industry Internship/ TBI Internship/Research Internship	0	0	24	12	50	50	100	-
TOTAL				**24 Hrs.**			**12**				

HONORS Specialization in Electronics and Computer Science
(AI for edge computing)

Sr. No.	Semester	Course Code	CourseTitle	Hours/ week			Credits	MaximumMarks			ESE Duration(Hrs.)
				L	T	P		ContinuousEvaluation	End Sem Exam	Total	
1	III	ECSTH3100	Fundamentals of Artificial intelligence and Edge Computing	3	0	0	3	50	50	100	3
2	IV	ECSTH4100	Single dimension signal processing on Edge	3	0	0	3	50	50	100	3
3	V	ECSTH5100	Image and Video Signal Processing on Edge	3	1	0	4	50	50	100	3
4	VI	ECSTH6100	Computer vision with Embedded Machine Learning	3	1	0	4	50	50	100	3
5	VII	ECSPH7100	Honors Project	0	0	8	4	50	50	100	-
				12	**2**	**8**	**18**				

MINOR Specialization in Electronics and Computer Science
(IoT)

Sr. No.	Semester	Course Code	Course Title	Hours per week			Credits	Maximum Marks			ESE Duration(Hrs)
								Continuous Evaluation	End Sem Exam	Total	
1	III	ECSTM3100	IoT fundamentals	3	0	0	3	50	50	100	3
2	IV	ECSTM4100	Sensor Interfacing with Arduino and ESP8266	3	0	0	3	50	50	100	3
3	V	ECSTM5100	Cloud Computing Using Raspberry Pi	3	1	0	4	50	50	100	3
4	VI	ECSTM6100	Data Management and Analytics for IoT	3	1	0	4	50	50	100	3

5	VII	ECSPM7100	Minor Project	0	0	8	4	50	50	100	-
			Total	**12**	**2**	**8**	**18**				

HONORS Specialization in Research

1	RM	ECST8004	Research Methodology	4	0	0	4	50	50	100	3
2	PRJ	ECSP8005	Research Internship	0	0	28	14	50	50	100	-
TOTAL				**4**	**0**	**28**	**18**				
				32 Hrs.							

Programme Electives

Micro Specialization		Semester V	Semester VI		Semester VII	Semester VIII	
		Elective-I	**Elective-II**	**Elective-III**	**Elective-IV**	**Elective-V**	**Elective-VI**
AI/ML	**Course Code**	ECST5005-1/ ECSP5005-1	ECST6003-1/ ECSP6003-1	ECST6004-1/ ECSP6004-1	ECST7005-1	ECST8001-1	ECST8002-1
	Course Name	Image Processing	Deep Learning-I	Natural Language Processing	Deep Learning-II	Generative Adversarial Network	Reinforcement Learning
VLSI	**Course Code**	ECST5005-2/ ECSP5005-2	ECST6003-2/ ECSP6003-2	ECST6004-2/ ECSP6004-2	ECST7005-2	ECST8001-2	ECST8002-2
	Course Name	VLSI Signal Processing	C Based VLSI Design	Design for Testability	Advanced Computer Architecture	Flexible Electronics and Sensors	Nano Electronics
IoE	**Course Code**	ECST5005-3/ ECSP5005-3	ECST6003-3/ ECSP6003-3	ECST6004-3/ ECSP6004-3	ECST7005-3	ECST8001-3	ECST8002-3
	Course Name	IoT Sensors and Devices	IoT Networks and Protocols	IoT Programming and BigData	Cyber Security and Privacy in IoT	Autonomous Vehicle	Capstone Project
General	**Course Code**	ECST5005-4/ ECSP5005-4	ECST6003-4/ ECSP6003-4	ECST6004-4/ ECSP6004-4	ECST7005-4	ECST8001-4	ECST8002-4
	Course Name	Cloud Computing	Data Mining and Warehousing	Big Data Web Intelligence	System Design	Block Chain	Sales Force

List of Open Electives

Sr. No.	Semester	Course Code	Course Name
1	III	ECST2980	Basics of Linux Operating System
2	IV	ECST2990	Designing with Raspberry Pi
3	V	ECST3980	Programming for Vedic Mathematics Sutras

Course Code	PHT1007				
Category	Basic Science Course				
Course Title	Physics				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	I

Course Outcomes

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

1. Outline the difference between intrinsic/extrinsic semiconductors and their carrier transport phenomena in semiconductors.
2. Illustrate the working and design aspects for various photonic devices like LEDs, solar cells, and LASER diodes.
3. Classify materials on the basis of band theory and its importance for semiconductors.
4. Apply fundamental knowledge of quantum mechanics to examine electron behavior in solids at the quantum level.
5. Analyze the process of generation and recombination of excess charge carriers in semiconductors along with working principle of P-N junction and Metal-Semiconductor junction diode.

Syllabus

Module 1: Introduction to Quantum Mechanics
Wave-particle duality, Heisenberg uncertainty relations, the quantum state wave function and its probability interpretation, Schrödinger's equation, Particle in an infinite potential well, Quantum tunneling.

Module 2: Electronic Materials
Formation of energy bands in solids, Classification of electronic materials, Kronig-Penney model, E-k diagram, Direct and indirect bandgaps, Valence and conduction bands, Density of states, Fermi-Dirac statistics, Fermi level, Effective mass.

Module 3: Intrinsic and Extrinsic Semiconductors
Intrinsic and extrinsic semiconductors, Dependence of Fermi level on carrier concentration and temperature, Carrier transport: diffusion and drift.

Module 4: Non-Equilibrium Semiconductors
Carrier generation and recombination, Continuity equation, Ambipolar transport equation, Quasi-Fermi Energy levels, Excess Carrier Lifetime, Qualitative introduction to recombination mechanisms.

Module 5: Junction Physics
p-n junction diode, Zero-applied bias, forward bias, reverse bias, Application of diode as a rectifier, Zener diode, Special diodes: Tunnel diode, Schottky diode, Ohmic contacts, NPN and PNP transistor and its characteristics, classification.

Module 6: Optoelectronic Devices
Optical absorption in semiconductors, Light emitting diodes, Laser diode, Stimulated emission and photon amplification, Einstein Coefficients, Solar Energy Spectrum, Solar cells.

Textbook(s)

1. *Semiconductor Physics and Devices* (Fourth Edition), Donald A. Neamen, McGraw-Hill, 2012.
2. *Semiconductor Device Physics and Design*, Umesh K. Mishra and Jasprit Singh, Springer, 2008.
3. *Electronic Devices and Circuits*, Jacob Millman, Christos C. Halkias, McGraw-Hill, 1967.

References

1. *Optoelectronics and Photonics: Principles and Practices* by S. O. Kasap, Prentice Hall, 2001.
2. *Physics of Semiconductor Devices*, Simon M. Sze, Wiley-Interscience, 1981.

Course Code	PHP1007				
Category	Basic Science Course				
Course Title	Physics Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

1. Develop skills required for experimentation and verification of physics laws.
2. Analyse the results obtained through proper graph plotting and error analysis.
3. Conduct experiments to validate physical behaviour of materials/components.
4. Analyse the behaviour and characteristics of P-N Junction, Zener Diode, and other semiconductor devices.
5. Prepare laboratory reports on interpretation of experimental results.

\

List of Experiments

1. Parameter extraction from V-I characteristics of a diode
2. Parameter extraction from V-I characteristics of a transistor
3. Analysis of diode rectifier
4. Resistivity measurement of semiconductor by Four Probe method
5. Performance and analysis of Hall Effect in semiconductor to determine the Hall coefficient and carrier concentration of the majority carriers in the given specimen
6. Estimation of energy gap in semiconductor
7. Characteristics and analysis of solar cells
8. Verification of Ohm's law and error analysis of the data using Linear Least Square Fit (LLSF) method
9. Analysis of energy values and wavefunction using Mathematica software
10. Verification of Planck's constant

Reference: Laboratory Manual of the Physics Department, RCOEM.

Course Code	MAT1001				
Category	Basic Science Course				
Course Title	Applied Mathematics-I				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	I

Course Outcomes

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

1. Recognize first order ordinary differential equations that can be solved by each of the four methods –Linear DE, Exact DE, reducible to linear DE and reducible to exact differential equations and use the appropriate method to solve them.
2. Solve higher order ordinary differential equations with constant and variable coefficients.
3. Find best fit curve by method of least square method and calculate correlation, regressions.
4. Recognize and understand discrete, continuous probability distributions and apply Binomial distribution, Poisson distribution and Normal distribution to appropriate problems.
5. Internalize multivariable calculus and apply it find Jacobians, maxima and minima of function/Solve numerical integrations by Newton coatformulas and Gauss-Legendre Quadrature.

Syllabus

Module 1: First order ordinary differential equations
Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut'stype, Applications of First order Differential Equations.

Module 2: Ordinary differential equations of higher orders
Second order linear differential equations with constant and variable coefficients, method of variation of parameters, Cauchy-Euler equation. Applications of Higher order Differential Equations.

Module 3: Statistics
Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves, correlation and regression – Rank correlation, Multiple regression and correlation and its application in Engineering.

Module 4: Differential Calculus
Taylor's and Maclaurin's series expansions, radius of curvature (Cartesian form), evolutes and involutes, Limit and continuity of functions of several variables and their partial derivatives, Eulers Theorem, chain rule, total derivative, Jacobians, Maxima, minima and saddle points; Method of Lagrange multipliers.

Module 5: Probability (For All Branches except Mechanical Branch)
Probability spaces, conditional probability, independence, Bay's Theorem, Discrete random variables, Binomial distribution, Poisson distribution, Normal distribution. Relation between binomial, Poisson and Normal distributions.

OR

Module 5: Numerical Integration (Only for Mechanical Branch)
Simpson's 1/3rd rule, 3/8th rule, Trapezoidal rule, Gauss-Legendre Quadrature.

Textbooks/References

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. W. E. Boyce and R. C. DiPrima, Elementary Differential Equations and Boundary Value Problems, 9th Edition, Wiley India, 2009.
3. S. L. Ross, Differential Equations, 3rd Ed., Wiley India, 1984.
4. E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India, 1995.
5. E. L. Ince, Ordinary Differential Equations, Dover Publications, 1958.
6. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
7. Theory and Problems of probability and statistics : 2nded :J. R. Spiegal ,Schaum series
8. A text book of Applied Mathematics Volume I & II, by P. N. Wartikar and J. N. Wartikar, Pune VidhyarthiGrihaPrakashan, Pune-411030 (India).
9. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.

Course Code	MAP1001				
Category	Basic Science Course				
Course Title	Computational Mathematics Lab				
Scheme & Credits	L	T	P	Credit	Semester
	0	0	2	1	I

Course Outcomes

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

1. Download SageMath and use it as an advanced calculator.
2. Sketch and analyze function graphs.
3. Apply the concepts of differential calculus to find extreme value of continuous functions and analyze solutions of differential equations.
4. Evaluate improper integrals and its applications to find length, area, volume, centre of gravity and mass.
5. Analyze and calculate eigenvalues, eigenvectors, rank nullity, and solve system of linear equations of a matrix/linear map.
6. Analyze the data to find best fit curve.

List of Experiments

1. To use SageMath as advanced calculator
2. 2D Plotting with SageMath
3. 3D Plotting with SageMath
4. Differential Calculus with SageMath
5. Solution of differential equations in SageMath
6. Basics of Linear Algebra
7. Curve Fitting by using SageMath
8. Integral Calculus with SageMath

Course Code	CHP1002				
Category	Basic Science Course				
Course Title	Environmental Science Lab				
Scheme & Credits	L	T	P	Credit	Semester
	0	0	2	1	I

Course Outcomes

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

1. Estimate different physicochemical parameters of water as well as air and their analysis methods
2. Explain the uses of greener and sustainable routes for the synthesis of biofuels/nano-materials
3. Determine nutrients/pollutants in soil sample

List of Experiments

1. Determination of pH, turbidity and TDS present in given water/wastewater sample
2. Determination of conductivity present in given water/wastewater sample
3. Determination of Dissolved Oxygen content in various natural sources of water/wastewater sample
4. Determination of Chemical Oxygen Demand (COD) present in various natural sources of water/wastewater sample
5. Determination of free residual chlorine in water/wastewater sample
6. Determination of concentration of acid in the beverage
7. Identification of microplastics in the given water sample
8. Estimation of Copper metal from e-waste
9. Green synthesis of Biodiesel (transesterification of oil)
10. Green synthesis of Metal nanoparticles by using various plant extract
11. Demonstration on measuring ambient air quality by using Air Quality Analyzer
12. Determination of particulate matter in the air using a High Air Volume Sampler
13. Identification of Alkali/Alkaline earth metal in water/wastewater using flame photometer
14. Determination of Organic Carbon / Total Iron / Total Nitrogen / Phosphorous / Potassium / Sulphur in Soil sample

Text Books

1. A Textbook on Experiments and Calculations in Engineering Chemistry by S.S. Dara, S. Chand Publications
2. Advanced Practical Physical Chemistry by J.B. Yadav, Krishna's Prakashan Media (P) Limited

Reference Books

1. Collection of Interesting General Chemistry Experiments, by A.J. Elias, Universities Press Publications
2. College Practical Chemistry by V.K. Ahluwalia, S. Dhingra and A. Gulati, Universities Press Publications
3. Standard Methods for the Examination of Water and Wastewater, American Public Health Association, American Water Works Association, Water Environment Federation

Course Code	ECST1001				
Category	Engineering Science Course				
Course Title	Programming for Problem Solving				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	I

Course Outcomes

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

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

Syllabus

Module 1: Introduction to Programming
Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.) Idea of Algorithm: Steps to solve logical and numerical problems. Representation of Algorithm: Flowchart / Pseudocode with examples. Arithmetic expressions and precedence.

Module 2: C Programming Language
Introduction to C language: Keywords, Constant, Variable, Data types, Operators, Types of Statements, Pre-processor Directives, Decision Control Statement – if, if-else, nested if-else statement, switch case, Loops and Writing and evaluation of conditionals and consequent branching.

Module 3: Arrays and Basic Algorithms
Arrays: 1-D, 2-D, Character arrays and Strings. Searching, Basic Sorting Algorithms, Finding roots of equations, notion of order of complexity through example programs (no formal definition required).

Module 4: Functions and Recursion
User defined and Library Functions, Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference. Recursion: As a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series.

Module 5: Pointers and Structures
Structures, Defining structures, Array of Structures, Introduction to pointers, Defining pointers, Pointer arithmetic, pointer operators, Use of Pointers in self-referential structures.

Module 6: File Handling
Streams in C, Types of Files, File Input / Output Operations: Modes of file opening, Reading and writing the file, Closing the files using fflush().

Text Books

1. Programming in ANSI C: E. Balguruswami, McGraw Hill
2. Mastering C: K.R. Venugopal and S.R. Prasad, Tata McGraw Hill

Reference Books

1. Programming with C: Byron Gottfried, Schaum's Outline Series
2. Let Us C: Yashwant Kanetkar, BPB Publication

Course Code	ECSP1001				
Category	Engineering Science Course				
Course Title	Programming for Problem Solving Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

List of Experiments

1. Using basic data types of C, implement arithmetic expressions.
2. Implement Programs using Decision Control Structures
3. Demonstrate use of Loop Control Structures
4. Implement Programs using Multi-way Decision Control Structures (Switch Case)
5. Apply Functions and Recursion to simplify programs.
6. Initialize array and apply it to solve problems of 1D and 2D arrays.
7. Demonstrate use of Structures and Pointers.
8. Apply file handling concepts in C.

Note: 2/3 Practice Programs will be taken on each of the experiments mentioned above.

Course Code	ECST1002				
Category	Engineering Science Course				
Course Title	Fundamentals of Computer Engineering				
Scheme & Credits	L	T	P	Credits	Semester
	2	0	0	2	I

Course Outcomes

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

1. Understand the basic components of computer systems and its functionality.
2. Understand data representation and the storage of data within a computer system.
3. Understand how computers can be linked together to share information and resources using worldwide network of networks.
4. Describe the functions of operating system and its role as a resource manager to execute any application.
5. Explore use cases and applications of AI, understand AI concepts and terms like machine learning, deep learning and neural networks.

Syllabus

Module 1
Introduction to Computer Science: Role of Algorithms, History of Computing, Science of Algorithms, Abstractions. Basics of data encoding and storage: Bits and their storage, Number system, Main memory, Mass Storage, Representing Information as Bit Patterns. Machine Architecture: CPU Basics, Stored Program concepts, Machine Language Introduction with example, Program Execution with illustrative example.

Module 2
Operating Systems: History of OS, OS Architecture, Coordinating Machine Activities. Networking and the Internet: Network Fundamentals, the Internet, the World Wide Web. Software Engineering: Introduction, Software Life Cycle. Database Systems: Database Fundamentals, Relational Model.

Module 3
Introduction to Artificial Intelligence: Intelligence and Machines, Perception, Reasoning, Machine Learning, Artificial Neural Networks, Deep Learning. Computer Graphics: Scope of Computer Graphics.

Text Books

1. Brookshear J.G., “Computer science: an overview”, Eleventh Edition, Addison-Wesley Publishing Company; 2011.

Reference Books

1. Silberschatz A., Gagne G., Galvin P.B., “Operating system concepts”, Ninth Edition, Wiley, 2012.
2. Cobbaut P., “Linux Fundamentals”, Samurai Media Limited, 2016.
3. Silberschatz A, Korth H.F, Sudarshan S., “Database system concepts”, Sixth Edition, McGraw Hill, 2010.
4. Kurose J.F., Ross K.W., “Computer networking: a top-down approach”, Sixth Edition, Pearson, 2013.
5. Peter Norvig and Stuart J. Russell, “Artificial Intelligence: A Modern Approach”, Pearson, Third edition, 2010.

Course Code	ECST1003				
Category	Programme Core Course				
Course Title	Electronics and Computer Workshop				
Scheme & Credits	L	T	P	Credits	Semester
	1	0	0	1	I

Course Outcomes

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

1. Inspect techniques to identify and test different Electronic components and Integrated Circuits.
2. Comprehend different EDA tools required for designing of Electronic and computer related circuits.
3. Classify mounting and troubleshooting practices and OS installation.
4. Understand data visualization and business intelligence tools.

Syllabus

Module 1 Basic Electronic Components Testing and Measurement
Basic electronic components, data sheets, electronic test and measurement equipments: multi meter, Cathode Ray Oscilloscope (CRO), Digital storage oscilloscope (DSO), function generator, power supply, spectrum analyzer etc.

Module 2 Switches (Electronic and Network)
Types of Switches – Construction, Working, Characteristics & Applications

Module 3 Internet of Things (IoT) Sensors and Actuators
Types, Characteristics & Applications

Module 4 Computer Hardware and Software
Assembling and disassembling CPU and identification of peripherals, Processor mounting and troubleshooting practices, USB, Ethernet, HDMI, thunderbolt port variants (peripherals), Types of OS and OS installation, OS imaging.

Module 5 Data visualization and business intelligence tools: Power BI and Tableau
Features, Architecture, Data Modeling, Dashboard Design, Data Sources

Course Code	ECSP1003				
Category	Programme Core Course				
Course Title	Electronics and Computer Workshop Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

List of Experiments:

1. Identification and testing of Resistors with the help of Color-Bands & by use of Multimeter and verify difference in the values.
2. Identify the various types of Capacitances and Find out the values using Color Code/written values on them.
3. Identify the terminals of a Diode and its Polarity, Identify the terminals of a Transistor and its Type (NPN or PNP)
4. Identify the various tools & write down their uses. Material required: various tools - Wire Cutter, Wire Stripper - Various types of Pliers - Vice - Crimping Tools (RJ-11/RJ-45) - Screw-Driver
5. Identify the various type of connectors used in various Gadgets & Instruments/Equipments.
6. Solder the joint connection of wires and check it. De-soldering and Re-soldering
7. Familiarization of function generator & Power Supply and perform measurements using it.
8. Learn cathode ray oscilloscope, DSO and perform measurements.
9. Assembling of electronic circuits using SMT (Surface Mount Technology) stations
10. Study of peripherals of a computer and its functions.
11. Assembling and disassembling of PC
12. Installation of Operating Systems – Windows 30
13. Installation of Operating Systems – LINUX 36
14. Hardware Troubleshooting and Software Troubleshooting
15. Understand modern application development A • HTML B • Wordpress C • Drupal
16. Familiarization and application of data visualization and business intelligence tools: Power BI and Tableau
17. Project

Text Books

1. K. A. Navas; Electronics lab Manual; Fifth Edition; PHI learning; 2015
2. N. Kumar, T. H. Sheikh; PC Assembly and Installation; Books clinic Publishing; 2020

Reference books

1. C. Bhargava; Digital Electronics: A Comprehensive Lab Manual; BSPublication; 2019

2. C. Zacker; PC Hardware: The Complete Reference; First Edition; McGraw Hill Education; 2017

Course Code	HUT1002				
Category	HSSM-AEC				
Course Title	English for Professional Communication				
Scheme & Credits	L	T	P	Credits	Semester
	2	0	0	2	I

Course Outcomes

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

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

Syllabus

Module 1 Vocabulary Building
Importance of using appropriate vocabulary
Techniques of vocabulary development
Commonly used power verbs, power adjectives and power adverbs.
Synonyms, antonyms, phrases & idioms, one-word substitutions and standard abbreviations

Module 2 Listening and Reading Comprehension
Listening Comprehension: active listening, reasons for poor listening, traits of a good listener, and barriers to effective listening
Reading Comprehension: types and strategies.

Module 3 Functional Grammar and Usage
Identifying Common Errors in use of: articles, prepositions, modifiers, modal auxiliaries, redundancies, and clichés
Tenses
Subject-verb agreement, noun-pronoun agreement
Voice

Module 4 Writing Skills
Sentence Structures
Sentence Types
Paragraph Writing: Principles, Techniques, and Styles

Module 5 Writing Practices
Art of Condensation: Précis, Summary, and Note Making
Correspondence writing techniques and etiquettes – academic writing
Essay Writing

Reference Books

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

Course Code	HUP1002				
Category	HSSM-AEC				
Course Title	English for Professional Communication Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

List of Practical's

Computer Assisted and Activity Based Language Learning

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

Activity Based Language Learning

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

Course Title: Liberal/Performing Arts

Course Code	Course Name	Sem.	Hours/ week	Credits	Continuous EvaluationMax. marks
HUP0001-1	Fundamentals of Indian Classical Dance: Bharatnatayam	I	2	1	50
HUP0001-2	Fundamentals of Indian Classical Dance: Kathak	I	2	1	50
HUP0001-3	Introduction to Digital Photography	I	2	1	50
HUP0001-4	Introduction to Japanese Language and Culture	I	2	1	50
HUP0001-5	Art of Theatre	I	2	1	50
HUP0001-6	Introduction to French Language	I	2	1	50
HUP0001-7	Introduction to Spanish Language	I	2	1	50
HUP0001-8	Art of Painting	I	2	1	50
HUP0001-9	Art of Drawing	I	2	1	50
HUP0001-10	Nature camp	I	2	1	50
PEP0001-21	Disaster Management through Adventure Sports	I	2	1	50
PEP0001-22	Self-defense Essentials and Basics Knowledge of Defense forces	I	2	1	50
CHP0001-31	Art of Indian traditional cuisine	I	2	1	50
CHP0001-32	Remedies by Ayurveda	I	2	1	50

Course Code	HUP0001-1				
Category	Co-Curricular Activity				
Course Title	Fundamentals of Indian Classical Dance: Bharatnatyam				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Recommended Reading

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

Course Code	HUP0001-2				
Category	Co-CurricularActivity				
Course Title	Fundamentals of Indian Classical Dance: Kathak				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

1. Orientation in Kathak. Correct posture of Kathak, Basic Movements and exercise Stepping, Chakkar of 5 count (Bhramari)
2. Practice sessions of practical 1
3. Hastaks, Hastaks and Steppings, Reciting a samyukta Mudra shloka, Hastak and steppings
4. Practice sessions of practical 3
5. Todas and Asamyukta hasta mudra shlok, Vandana of Shlok, 2 Todas and Vandana, Ghante Ki Tihai
6. Practice sessions of practical 5
7. 21 Chakkardar Toda and Ginnti Ki Tihai, 2 Todas and 1 Chakkardar Toda, practice sessions
8. Final performances.

Recommended Reading

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

Course Code	HUP0001-3				
Category	Co-Curricular Activity				
Course Title	Introduction to Digital Photography				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

1. Develop an understanding of the technical aspects and aesthetics of Photography.
2. Apply the rules of digital photography for creating photographs.
3. Develop skills to enhance photographs through post processing.
4. Create a portfolio of their photographs in selected genre.

Syllabus

1. Orientation in digital photography: Genres, camera handling and settings
2. Rules of Composition
3. Rules of Composition: practice sessions
4. Understanding Exposure and Art of Pre-Visualization
5. Rules of Composition and Art of Pre-Visualization: practice sessions
6. Post Processing Photographs and Portfolio creation
7. Post Processing Photographs: practice sessions
8. Portfolio finalization and presentation in selected genre.

Reference material

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

Course Code	HUP0001-4				
Category	Co-Curricular Activity				
Course Title	Introduction to Japanese Language and Culture				
Scheme &Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Recommended reading

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

Course Code	HUP0001-5				
Category	Co-Curricular Activity				
Course Title	Art of Theatre				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Reference books

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

Course Code	HUP1003-6				
Category	Co-Curricular Activity				
Course Title	Introduction to French Language				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

List of Practicals

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

Recommended Reading

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

Course Code	HUP0001-7				
Category	Co- Curricular Activity				
Course Title	Introduction to Spanish Language				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

List of Practicals

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

Recommended Reading

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

Course Code	HUP0001-8				
Category	Co- Curricular Activity				
Course Title	Art of Painting				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Reference Material

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

Course Code	HUP0001-9				
Category	Co- Curricular Activity				
Course Title	Art of Drawing				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Syllabus

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

Reference Material

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

Course Code	HUP0001-10				
Category	Co-Curricular Activity				
Course Title	Nature Camp				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

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

Course Content

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

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

Course Code	PEP0001-21				
Category	Co-Curricular Activity				
Course Title	Disaster Management Through Adventure Sports				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

On completion of the course, students will be able to

1. Understand the meaning and importance of adventure sports.
2. Learn the various types of adventure sports, the equipment and resources required to practice disaster management activities.
3. Learn the safety measures related to different risks and their management.
4. To Apply disaster management theory to institutional and societal problems and situations.

Syllabus

1. Basic adventure
2. First Aid
3. Various types of knots
4. Shelter making
5. Disaster management
6. Team building and goal setting
7. Realization of fear, risk and their roles and analyzing safety management plan

Course Code	PEP0001-22				
Category	Co-Curricular Activity				
Course Title	Self-defense Essentials and Basics Knowledge of Defense forces				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

1. Understand the meaning, need and fitness requirements to implement self-defense
2. Learn the basic techniques of selected combative sports.
3. Learn to prepare basic Physical Training for Defense forces.
4. Implement survival techniques during emergencies.

Syllabus

1. General conditioning and self-defense specific conditioning
2. Applications of techniques of combative sports for self-defense.
3. Self-defense techniques for specific situations: chain snatching, knife or stick attack, holding from back or front etc.
4. Basic Military Knowledge and exposure making students Confident, bold, disciplined and trains them to join Armed Forces.

Course Code	CHP0001-31				
Category	Co- Curricular Activity				
Course Title	Art of Indian traditional cuisine				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

1. Understand the factors that affect regional eating habits and the unique ingredients found in various states of India
2. Get insight to prepare popular dishes from various regions of India.

Module 1. Indian Regional foods and snacks - factors effecting eating habits.
Module 2. Indian gravies – ingredients, their importance
Module 3. Indian Sweets - ingredients, their importance
Module 4. Presentation of Indian Meals, Menu Planning, Food Costing
Module 5. Food Preservatives and Safety

List of experiments

1. Introduction to cookery: does and don'ts
2. Introduction to Indian cuisine, philosophy and classification.
3. Regional influence on Indian Food - factors affecting eating habits
4. Preparation of Garam masala and/or Chat masala with ingredients and their importance
5. Preparation of different gravies such as white, yellow or brown gravies with ingredients and their importance
6. Preparation of Indian sweets like Besan keladdu with ingredients and their importance
7. Presentation of meal, Menu planning and Food costing
8. Common chemical food preservatives and their safety standards.

Reference books

1. Arora, K.,; Theory of cookery; First Edition, Frank Brothers Company (Pub) Pvt.Ltd., 2008 ISBN: 9788184095036, 8184095031
2. Philip, Thangam.E.,; Modern Cookery: Vol.1; Sixth Edition, Orient Black Swan., 2008 ISBN: 9788125040446, 8125040447
3. Parvinder S; Quantity Food Production Operations and Indian Cuisine (Oxford Higher Education); First Edition; Oxford University Press, 2011 ISBN 10: 0198068492 ISBN 13: 9780198068495
4. Singh, Yogesh; A Culinary Tour of India; First Edition I.K. International Publishing House Pvt. Ltd. ISBN 978-93-84588-48-9

5. Singh Shakesh; Simplifying Indian Cuisine; First Edition, Aman Publications, ISBN 81-8204-054X
6. Dubey Krishna Gopal; The Indian Cuisine; PHI Learning Pvt. Ltd. ISBN 978-81203-4170-8

Course Code	CHP0001-32				
Category	Co- Curricular Activity				
Course Title	Introduction to Remedies by Ayurveda				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	I

Course Outcomes

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

1. Know basic principle of Ayurvedic formulations.
2. Different types of Natural Remedies.
3. Basic idea about their Characterization.

Module 1. Introduction to Ayurveda
Module 2. Different types of Ayurvedic formulations: Churn, Bhasma, Vati, Tailum
Module 3. Introduction to Methods of preparation
Module 4. Characterization, applications

Practical's based on above syllabus

1. Preparations of some medicinal oils like Bramhitel, Bramhi Awala, Vatnashak Tel, Bhurngraj Tel etc.
2. Preparation of Churn, like Trifala Churn, Hingastak Churn, Trikut Churn etc.
3. Preparation of some Bhasmas and vati

Books

1. Chemistry and Pharmacology of Ayurvedic Medicinal Plants by Mukund Sabnis, Chaukhambha Amarbharati Prakashan.
2. Everyday Ayurveda by Shailesh Rathod
3. A text Book of Rasashastra by Vikas Dhole and Prakash Paranjpe
4. A text Book of Bhaiajya Kalpana Vij¯nana

Course Code	HUT1004				
Category	HSSM-VEC				
Course Title	Foundation Course In Universal Human Value				
Scheme & Credits	L	T	P	Credits	Semester
	1	0	0	1	I

Course Outcomes

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

1. Develop a holistic perspective of life
2. Better understanding of inter-personal relationships and relationship with society and nature.
3. An ability to strengthen self-reflection

Syllabus

Module 1: Aspirations and concerns
Need for Value Education: Guidelines and content of value education.
Exploring our aspirations and concerns: Knowing yourself, Basic human aspirations
Need for a holistic perspective, Role of UHV;
Self-Management: harmony in human being

Module 2: Health
Harmony of the Self and Body, Mental and physical health;
Health for family, friends and society.

Module 3: Relationships and Society
Harmony in relationships, Foundational values: Trust, Respect, Reverence for excellence, Gratitude and love;
harmony in society; harmony with nature.

Reference Material
The primary resource material for teaching this course consists of

1. Textbook: R.R. Gaur, R. Sangal, G. P. Bagaria, *A foundation course in Human Values and professional Ethics*, Excel books, New Delhi, 2010, ISBN 978-8-174-46781-2

Reference books

1. B. L. Bajpai, 2004, *Indian Ethos and Modern Management*, New Royal Book Co., Lucknow. Reprinted 2008.
2. P. L. Dhar, R. R. Gaur, 1990, *Science and Humanism*, Commonwealth Publishers.
3. Sussan George, 1976, *How the Other Half Dies*, Penguin Press. Reprinted 1986, 1991
4. Ivan Illich, 1974, *Energy & Equity*, The Trinity Press, Worcester, and HarperCollins, USA
5. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, *Limits to Growth*, Club of Rome's Report, Universe Books.
6. Subhas Palekar, 2000, *How to practice Natural Farming*, Pracheen (Vaidik) Krishi Tantra Shodh, Amravati.
7. A. Nagraj, 1998, *Jeevan Vidya ek Parichay*, Divya Path Sansthan, Amarkantak.
8. E. F. Schumacher, 1973, *Small is Beautiful: a study of economics as if people mattered*, Blond & Briggs, Britain.
9. A. N. Tripathy, 2003, *Human Values*, New Age International Publishers.

Course Code	CHT2006				
Category	Basic Science Course				
Course Title	Chemistry of Smart Materials				
Scheme & Credits	L	T	P	Credits	Semester
	2	0	0	2	II

Course Outcomes

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

1. Classify and explain the different types of sensors for various applications.
2. Discuss unique properties of nano-materials to solve challenges in our life and applications in computational world.
3. Discuss how spectroscopic methods are used for qualitative and quantitative analysis.
4. Analyze the utilization of green computing technology for environmental issues.

Syllabus

Module 1: Smart Sensors and Materials
RFID and IONT materials: Synthesis, properties and applications in logistic information, intelligent packaging systems (Graphene oxide, carbon nanotubes (CNTs) and polyaniline).
Sensors: Introduction, types of sensors (Piezoelectric and electrochemical), nanomaterials for sensing applications (Strain sensors, gas sensor, biomolecules and volatile organic compounds).

Module 2: Nanomaterials
Introduction, classification, size dependent properties, surface area, optical and catalytic properties, Synthesis methods of nanomaterials - Top down and bottom-up approach.
Carbon nanomaterials: Types, properties and applications of CNT and graphene.
Applications of nanomaterials.

Module 3: Characterization Techniques and Computational Tools
Fundamentals of spectroscopy, Electronic Spectroscopy, Nuclear Magnetic Resonance Spectroscopy.
Basics of Nuclear magnetic resonance quantum computer.
Synthesis of drugs, basic softwares for bio-chemical assessment of drugs.

Module 4: Green Computing and Chemistry
E-wastes - Types, environmental and health risks, segregation and recycling (Hydrometallurgical, pyrometallurgical and direct recycling),
Extraction of precious metals from e-wastes, Twelve principles of Green Chemistry.
Green Computing, Role of Green Computing in Environment and Research, Green devices and Green data Servers.

Text Books

1. Shikha Agrawal, *Engineering Chemistry: Fundamentals and Applications*, Cambridge University Press.
2. Dr. Rajshree Khare, *A Textbook of Engineering Chemistry (AICTE)*, S.K. Kataria & Sons.
3. S. S. Dara, *A Textbook of Engineering Chemistry*, S. Chand Publications.
4. A. K. Das and M. Das, *An introduction to nanomaterials and nanoscience*, CBS Publishers and Distributors.
5. M. Afshar Alam, Sapna Jain, Hena Parveen, *Green Computing Approach Towards Sustainable Development*, Wiley Interscience Publications.
6. *Sensor & Transducers*, D. Patranabis, 2nd edition, PHI.

Reference Books

1. *E-waste recycling and management: present scenarios and environmental issues*, Khan, Anish, and Abdullah M. Asiri, 2019, Springer, Vol. 33. ISBN: 978-3-030-14186-8.
2. Hans-Eckhardt Schaefer, *Nanoscience: The Science of the Small in Physics, Engineering, Chemistry, Biology and Medicine*, Springer-Verlag Berlin Heidelberg.

Course Code	CHP2006				
Category	Basic Science Course				
Course Title	Chemistry of Smart Materials Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	II

Course Outcomes

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

1. Apply the fundamental principles of measurement and skills in preparation and handling of hazardous chemicals and interpret the statistical data related to measurements.
2. Estimate the rate constants of reactions and order of the reaction and/or to validate adsorption isotherms.
3. Use of various computational tools for analysis of different spectral properties and bio-activities.

List of Experiments

1. Preparation of different Solutions: Molar solution, Normal solution and percent solution and Determination of concentration.
2. Demonstration of Handling of hazardous chemicals, MSDS (Material Safety Data Sheet), waste minimization strategies and chemical waste disposal.
3. Basic statistical analysis of results of neutralization of acid against the base and preparing acceptable graphs using software.
4. Prediction of infrared/NMR spectral and analytical data of organic molecules using Computational Software.
5. Spectroscopic/Colorimetric determination of wavelength of maximum absorption of chemical/biological compound in solution and determination of concentration using Lambert-Beer's Law.
6. To study chemical kinetics of peroxydisulphate and iodide ions reactions and to find out order of the reaction and analysis of experimental data using Computational Software.
7. Molecular docking of drugs using open computational software. (Demonstration experiment)
8. Determination of rate of the reaction at room temperature and analysis of experimental data using Computational Software.
9. Use of open access software for the interpretation of various parameters of materials including drugs.
10. Estimation of Copper from PCB.

Reference Books

1. S. S. Dara, *A Textbook on Experiments and Calculations in Engineering Chemistry*, S. Chand Publications.
2. J. B. Yadav, *Advanced Practical Physical Chemistry*, Krishna's Prakashan Media (P) Limited.
3. A. J. Elias, *Collection of Interesting General Chemistry Experiments*, Universities Press Publications.
4. V. K. Ahluwalia, S. Dhingra and A. Gulati, *College Practical Chemistry*, Universities Press Publications.
5. Ashutosh Kar, *Advanced Practical Medicinal Chemistry*, New Age International Publisher.

Reference Books

1. David Young, *Computational Chemistry: A Practical Guide for Applying Techniques to Real World Problems*, Wiley Interscience Publications.

Course Code	MAT2001				
Category	Basic Science Course				
Course Title	Applied Mathematics-II				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	II

Course Outcomes

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

1. Interpret the solutions of system of linear equations and use the concepts of Eigenvalues, Eigenvectors to find diagonalization of matrices, reduction of quadratic form to canonical form.
2. Evaluate definite and improper integrals using Beta, Gamma functions. Also trace cartesian curves.
3. Solve multiple integration by change of order, change of variable methods and apply it to find area, volume, mass and centre of gravity.
4. Understand geometric meaning of gradient, curl, divergence.
5. Perform line, surface and volume integrals of vector-valued functions / Analyze and compare different sets of data and classify the data by means of diagrams and graph.

Syllabus

Module 1: Matrices
Algebra of matrices, Inverse and rank of a matrix, rank-nullity theorem;
System of linear equations; Symmetric, skew-symmetric and orthogonal matrices;
Eigenvalues and eigenvectors; Diagonalization of matrices; Cayley-Hamilton Theorem, Orthogonal transformation and quadratic to canonical forms, Introduction to n-dimensional space.

Module 2: Integral Calculus
Evaluation of definite and improper integrals; Beta and Gamma functions and their properties;
Tracing of curves (Cartesian form)

Module 3: Multiple Integrals
Multiple Integration: Double and triple integrals (Cartesian and polar), change of order of integration in double integrals,
Change of variables (Cartesian to polar), Applications: area, mass and volume by double integration, Center of mass and Gravity (basic concepts).

Module 4: Vector Calculus (Differentiation)
Scalar point function, Vector point function, vector differentiation, gradient, divergence and curl, Directional derivatives with their physical interpretations, solenoidal and irrotational motions, Scalar potential function.

Module 5: Vector Calculus (Integration) *(All Branches except Biomedical Engineering)*
Vector integration: Line integrals, work done, conservative fields, surface integrals and volume integrals,
Stoke's theorem, Gauss divergence theorem, Green's theorem and their simple applications.

OR

Module 5: Descriptive Statistics *(Only for Bio-Medical Engineering)*
Types of statistical data: categorical, ranked, discrete, and continuous.
Distinction between univariate, bi-variate, and multivariate statistics,
Visualization techniques such as joint contingency tables, scatter plots, 2D histograms and line graphs,
Measures of central tendency and Dispersion.

Topics for Self Learning
Rolle's theorem, Mean value theorems, Indeterminate forms, Applications of definite integrals to evaluate perimeter, area, surface areas and volumes of revolutions.

Textbooks / References

1. Erwin Kreyszig, *Advanced Engineering Mathematics*, 9th Edition, John Wiley & Sons, 2006.
2. Veerarajan T., *Engineering Mathematics for First Year*, Tata McGraw-Hill, New Delhi, 2008.
3. B. S. Grewal, *Higher Engineering Mathematics*, Khanna Publishers, 35th Edition, 2000.
4. Ramana B. V., *Higher Engineering Mathematics*, Tata McGraw Hill, New Delhi, 11th Reprint, 2010.
5. P. N. Wartikar and J. N. Wartikar, *A Text Book of Applied Mathematics Volume I & II*, Pune Vidhyarthi Griha Prakashan, Pune-411030 (India).
6. *Biomedical Statistics* - Shantikumar Yadav, Sompal Singh, Ruchika Gupta
7. *Theory and Problems of Probability and Statistics* - M. R. Spiegal (McGraw Hill), Schaum Series

Course Code	ECST2001				
Category	Engineering Science Course				
Course Title	Elements of IoT				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	II

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

1. Wire Raspberry Pi and create a fully functional computer
2. Use Python-based IDE and trace and debug Python code on the device
3. Measure physical parameter using sensors
4. Implement various communication protocols for wired and wireless communication
5. Interfaces different motors and create robots

Syllabus

Module 1
Basic functionality of the Raspberry Pi and Arduino and its Processor, setting and configuring the board, differentiating Raspberry Pi from other platforms like Arduino, Begal, Asus Thinker etc. Overclocking, Component overview.

Module 2
Introduction to Linux: Implications of an operating system on the behavior of the Raspberry Pi, Overview of Linux and its terminal command, apt-get update, apt-get upgrade, navigating the filesystem and managing processes, text-based user interface through the shell, overview of graphic user interface.

Module 3
Programming the Raspberry Pi:
Python: Introducing to Python programming language; Python Programming Environment, Python Expressions, Strings, Functions, Function Arguments, Lists, List Methods, Control Flow, Numpy, PIP (Python Installation Package) and customized libraries.
C++ programming: Basic C++ programming approach, header file structure and library organization, Cross Compiler and its configuration.

Module 4
Exploring Electronics with the Raspberry Pi:
Communication facilities on Raspberry Pi (I2C, SPI, UART), working with RPi.GPIO library, Interfacing of Sensors and Actuators.
Project 2: Set UP a Pi motion detector
Project 3: Set UP a Pi ADC/DAC
Project 4: CONSTRUCT a digital weather station
Project 5: CONSTRUCT a Traffic Light Controller

Module 5
Communication using Raspberry Pi: Wired and Wireless communication, TCP, IP configurations, SSH, Putty terminal usage.
Project 6: Set UP file server
Project 7: Network YOUR keyboard and MOUSE
Project 8: Create a portable wireless access point
Project 9: COMMUNICATE with ARDUINO
Project 10: CONSTRUCT a digital server-based weather station

Module 6
Robotic Motion PI: DC, Servo, Stepper, Motor Drivers, Motor Shields, Camera Interfacing, remote data logging.
Project 11: Keyboard Control Robot
Project 12: Wireless Robot

Text Books

1. *Raspberry Pi 3: An introduction to using with Python Scratch, Javascript and more*, Gary Mitnick, Create Space Independent Publishing Platform, 2017.
2. *Raspberry Pi for Python Programmers Cookbook*, Tim Cox, Packt Publishing Limited; 2nd Revised edition, 2016.
3. *Raspberry Pi User Guide*, Eben Upton and Gareth Halfacree, John Wiley & Sons, 2016.

Course Code	ECSP2001				
Category	Engineering Science Course				
Course Title	Elements of IoT Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	II

Course Outcomes

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

1. Wire Raspberry Pi and create a fully functional computer
2. Use Python-based IDE and trace and debug Python code on the device
3. Measure physical parameter using sensors
4. Implement various communication protocols for wired and wireless communication
5. Interfaces different motors and create robots

List of Experiments

1. LED control on Esp8266 programmed via Arduino IDE.
2. User identification using RFID chip RC522
3. Motor interfacing with Raspberry Pi.
4. Send SMS alert from RPi if obstacle detected.
5. To check door status on intialstate.com using RPi via internet
6. To send DHT11 temperature to Thingspeak using Raspberry Pi
7. Image tweeted when camera detects motion using Raspberry Pi.
8. Send data of various sensor to Thingspeak using NodeMCU
9. Display DHT11 sensor data on OLED screen using NodeMCU.
10. ESP8266 GPIO Control via Web Server programmed via Arduino IDE.

Course Code	ECST2002				
Category	Engineering Science Course				
Course Title	Object Oriented Programming				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	II

Course Outcomes

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

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

Syllabus

Module 1
Features of Object Oriented Programming languages, Abstraction, Encapsulation, Inheritance, polymorphism and late binding. Concept of a class, Access control of members of a class, instantiating a class, constructor and method overloading and overriding.

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

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

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

Module 5
Generics, generic class with two type parameter, bounded generics. Collection classes: Array list, Linked List, Hash set, Treeset, HashMap.

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

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

\

Text Books

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

Course Code	ECSP2002				
Category	Engineering Science Course				
Course Title	Object Oriented Programming Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	II

List of Experiments

1. Implement the Classes and Objects in Java.
2. Implement a program in Java with Constructors and Destructors. Also implement the concept of Overloading.
3. Demonstrate use of Inheritance.
4. Implement a concept of Interface in Java.
5. Demonstrate use of MultiThreading.
6. Implement Packages and import that package in program.
7. Demonstrate use of Exception Handling Mechanism.
8. Apply concept of Generics Class and Method.
9. Demonstrate Collection Framework and perform some basic operations on the ArrayList and HashSet.
10. Apply File Handling concepts in Java.

Course Code	ECST2003				
Category	Programme Core Course				
Course Title	Digital Electronics				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	II

Course Outcomes

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

1. Apply various optimization techniques to minimize digital circuits.
2. Design combinational logic circuits.
3. Analyze and design asynchronous and synchronous sequential circuits.
4. Discuss x86 architecture.

Syllabus

Module 1
Basics of Digital Electronics: Motivation for digital systems: Number Systems and arithmetics, Logic and Boolean algebra, logic gates & truth tables, SOP, POS, Minimization of combinational circuits using Karnaugh-maps.

Module 2
Combinational Circuit Design: Multiplexers, De-multiplexers, Encoders, Decoders, Code Converters, Adders, Subtractor (Half, Full), BCD Adder/Subtractor, ripple and carry look-ahead addition, Unsigned Multiplier.

Module 3
Sequential circuit Design-I: Storage elements, Flip-flops and latches: D, T, J/K, S/R flip-flops: level triggered, edge triggered, Master Slave flip-flop, flip flop conversion, timing analysis.

Module 4
Sequential circuit Design-II: Design of asynchronous and synchronous counters, Registers & Shift registers, Application of shift register: ring counter, Johnson counter, sequence generator and detector, serial adder; Linear feedback shift register (LFSR).

Module 5
Design of synchronous sequential circuit using Mealy model and Moore model: state transition diagram, algorithm state machine (ASM) chart.

Module 6
Introduction to X86 architecture.

Text Books

1. Donald P. Leach, Albert P. Malvino and Goutam Saha, “Digital Principles & Applications 8e”, McGraw Hill
2. Douglas V. Hall “Microprocessors and Interfacing”, Tata McGraw Hill Education Private Limited, 2005

Reference Books

1. Thomas L Floyd, “Digital Fundamentals 9e”, Pearson
2. M. Morris Mano and Michael D. Ciletti, “Digital Design 5e”, Pearson
3. Taub and Shilling, “Digital Integrated Electronics”, McGraw Hill
4. A Anand Kumar, “Fundamentals of Digital Circuits” Fourth Edition, PHI
5. Kip R. Irvine, “Assembly Language for x86 Processors” Seventh Edition, Pearson Education

Course Code	ECSP2003				
Category	Programme Core Course				
Course Title	Digital Electronics Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	II

List of Experiments

1. To verify truth table of different logic gates.

2. Design basic logic gates using universal gate and verify its truth table.

3. To verify the following Boolean expressions using gates and Multisim software:
   a) A+AB+ABA + AB + ABA+AB+AB
   b) AB(C+AC)AB(C + AC)AB(C+AC)

4. To implement the following arithmetic circuits using (a) logic gates IC's and (b) using Multisim software:
   a) Full adder
   b) Half subtractor

5. Implement the function F=Σm(1,3,5,7,8,9,11,13,15)F = \Sigma m (1,3,5,7,8,9,11,13,15)F=Σm(1,3,5,7,8,9,11,13,15) using 16:1 and 8:1 multiplexer.

6. Verify the truth table of SR, JK, JKMS, T and D flip flop.

7. To study the following functions of Shift register using IC 7495:
   a) SIPO (Serial In Parallel Out)
   b) PIPO (Parallel In Parallel Out)
   c) PISO (Parallel In Serial Out)
   d) SISO (Serial In Serial Out)

8. Design and verify 2-bit synchronous down counter using S-R flip-flop.

9. Design and verify the functionality of a sequence detector to detect the sequence 1101 using Mealy and Moore model and use J-K flip-flop to implement the design.

10. To study architecture, memory segmentation, and pipeline of 8086 microprocessor.

Course Code	HUT2001				
Category	Engineering Science Course				
Course Title	Foundational Literature of Indian Civilization				
Scheme & Credits	L	T	P	Credits	Semester
	2	0	0	2	II

Course Outcomes

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

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

Syllabus

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

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

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

Module 4 Indian wisdom through ages
Panchtantras, Purans: contents and issues of interests, Itihasa: uniqueness of the two epics (Ramayan and Mahabharata), Key issues and messages from Ramayana, Mahabharata – a source of worldly wisdom; Indian ancient Sanskrit literature: Kalidas, Vishakadutta, Bhavbhuti, Shudraka any one text as decided by the course teacher

Reference material

1. B. Mahadevan, Vinayak Rajat Bhar, Nagendra Pavana R. N., “Introduction to Indian Knowledge System: Concepts and Applications” PHI, 2022
2. S.C. Chatterjee and D.M. Datta, An introduction to Indian Philosophy, University of Calcutta, 1984

Course Code	PET2001				
Category	Basic Science Course				
Course Title	Sports-Yoga-Recreation				
Scheme & Credits	L	T	P	Credits	Semester
	1	0	0	1	I I

Course Outcomes

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

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

Syllabus

Module 1 Theory: Introduction

- Meaning, definition and importance of health & wellness
- Dimensions of health and wellness
- Factors influencing health and wellness
- Physical fitness, nutrition, habits, age, gender, lifestyle, body types
- Health & wellness through physical activities, sports, games, yoga and recreation activities
- Causes of stress & stress relief through exercise and yoga
- Safety in sport

References

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

Course Code	PEP2001				
Category	Basic Science Course				
Course Title	Sports-Yoga-Recreation Lab				
Scheme & Credits	L	T	P	Credit	Semester
	0	0	2	1	I I

Course Outcomes:

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

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

Module 1: - Practical – Exercises for Health and Wellness

• Warm-Up and Cool Down – General & Specific Exercises
• Physical Fitness Activities
• Stretching Exercises
• General & Specific Exercises for Strength, Speed, Agility, Flexibility, coordinative abilities
• Cardiovascular Exercises
• Assessment of BMI
• Relaxation techniques
• Physical Efficiency Tests

Module 2: - Yoga

• Shukshma Vyayam
• Suryanamaskar
• Basic Set of Yogasanas – Sitting, standing, supine and prone position
• Basic Set of Pranayama & Meditation

References:

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

Course Code	ECST3001				
Category	Programme Core Course				
Course Title	Data Structures				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	III

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

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

Syllabus

Module I: (5 Hours)
Understanding data structures and algorithms, Python for data, Variables and expressions, Flow control and iteration, Overview of data types, objects and Python modules, Types of Data Structures – User defined, Built-in data types: List, Set, Dictionary, Tuple.

Module II: (5 Hours)
Linear Data Structure – Arrays, Pointer structures, Nodes, Representation of arrays, Applications of arrays, Sparse matrix and its representation.

Module III: (4 Hours)
Stack: Stack – Definitions & Concepts, Operations on Stacks, Applications of Stacks
Queue: Representation of Queue, Operations on Queue, Applications of Queue
Linked List: Singly Linked List, Doubly Linked List, Circular Linked List, Linked implementation of Stack, Linked implementation of Queue, Applications of Linked List

Module IV: (6 Hours)
Nonlinear Data Structure:
Tree – Definitions and Concepts, Representation of binary tree, Binary tree traversal (Inorder, Postorder, Preorder), Binary search trees
Graph – Representation of Graphs, Elementary Graph operations, Breadth First Search, Depth First Search, Spanning Trees, Shortest path, Minimal spanning tree

Module V: (5 Hours)
Sorting and Searching – Insertion Sort, Quick Sort, Merge Sort, Heap Sort, Sorting on several keys, List and Table Sort, Linear Search, Binary Search

Module VI: (5 Hours)
Hashing and Symbol Tables – Perfect hashing functions, Putting elements, Getting elements, Testing the hash table, Non-string keys, Growing a hash table, Open addressing

Text Books

Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, *Data Structures and Algorithms in Python*, Wiley, 2013.

Reference Books

1. Gowrishankar S, Veena A, *Introduction to Python Programming*, 1st Edition, CRC Press/Taylor & Francis, 2019. ISBN-13: 978-0-8153-9437-2.
2. Benjamin Baka, *Python Data Structures and Algorithms*, Published by Packt Publishing Ltd., 2017.

Course Code	ECSP3001				
Category	Programme Core Course				
Course Title	Data Structures Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	III

List of Experiments

1. Write a Python program for class `Flower`, that has three instance variables of type `str`, `int`, and `float` that respectively represent the name of the flower, its number of petals, and its price. Your class must include a constructor method that initializes each variable to an appropriate value, and your class should include methods for setting the value of each type, and retrieving the value of each type.

2. Develop an inheritance hierarchy based upon a `Polygon` class that has abstract methods `area()` and `perimeter()`. Implement classes `Triangle`, `Quadrilateral`, `Pentagon`, that extend this base class, with the obvious meanings for the `area()` and `perimeter()` methods. Write a simple program that allows users to create polygons of the various types and input their geometric dimensions, and the program then outputs their area and perimeter.

3. Implement Method Overloading and Method Overriding.

4. Illustrate the following comprehensions using Python programming:
   a) List Comprehensions
   b) Dictionary Comprehensions
   c) Set Comprehensions
   d) Generator Comprehensions

5. Create a program to implement the binary search tree.

6. Develop a program that implements the Bubble Sort and Selection Sort algorithms.

7. Write a program to implement Merge Sort and Quick Sort.

8. Write a program to implement Stacks and Queues.

9. Write a program to implement Singly Linked List.

10. Write a program to implement Doubly Linked List.

Course Code	ECST3002				
Category	Engineering Science Course				
Course Title	Electronic Devices and Circuits				
Scheme and credits	L	T	P	Credits	Semester
	3	0	0	3	III

Course Outcomes

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

1. Identify the region of operation of PN Junction Diode, BJT and MOSFET.
2. Design rectifier, clipper, clamper, and voltage regulator using diodes.
3. Apply the mathematical models of BJT and MOS transistors for circuits and systems design.
4. Examine the effect of negative feedback on gain, bandwidth, input and output impedance and the stability of the amplifier.
5. Design, test and analyze operational amplifier-based circuits/systems.

Syllabus

Module I

Diode Models and Circuits: Terminal Characteristics of Junction Diodes, Models of P-N Junction Diode, Small Signal Model. Operation in the Reverse Breakdown Region—Zener Diodes, Zener as a Shunt Regulator, Applications of PN junction diode ─ Rectifier, Clipper, Clamper, DC power supply, Diode Logic Gates

Module II

Bipolar junction Transistor: Device structure and Physical Operation, Current Components in BJT, Transistor configurations and Input, Output characteristics, Load line concept, Biasing of BJT, Applications of BJT as a switch and single stage voltage amplifier.

Module III

MOS Field Effect Transistor: Device structure and physical operation, Current–Voltage Characteristics, MOSFET circuits at DC, MOSFET in Amplifier Design: The Voltage-Transfer Characteristic (VTC), biasing the MOSFET to Obtain Linear Amplification, Small-Signal Voltage Gain, Small-Signal Operation and Model.

Module IV

Feedback amplifier and Op-amp fundamentals: General Feedback amplifier Structure, Properties of Negative Feedback, Characteristics of operational amplifier, open loop Op-amp, basic Inverting and Non-inverting Op-amp amplifiers with negative feedback, Op-amp parameters & their analysis.

Module V

Op-amp linear and nonlinear applications: Voltage follower, summing amplifiers, integrators and differentiators, difference amplifiers & instrumentation amplifiers, Comparators, Schmitt trigger circuits, Sample/Hold circuits

Module VI

Oscillators and Active filters design: Precision rectifiers, oscillators: basic concept, Op-amp based sinusoidal oscillators, design of Active filters. Digital to analog converters, Analog to digital converters.

Textbooks

1. Adel S. Sedra, Kenneth C. Smith, Arun N. Chandorkar: Microelectronic Circuits: Theory and Applications: Seventh Edition, Oxford University Press, 2017.
2. Sergio Franco, “Design with Operational Amplifiers and Analog Integrated Circuits” Fourth Edition, McGraw-Hill Education, 2014.

Reference Books

1. Donald Neamen “Electronic Circuits: Analysis and Design” Third Edition, McGraw-Hill Publication
2. Ramakant Gayakwad, ”OP-AMPS and linear integrated circuits” 4th Edition, PHI
3. Jacob Millman, Christos Halkias, Chetan Parikh: “Millman's Integrated Electronics” Second edition, McGraw Hill Education, 2017.
4. Coughlin Driscoll, “Operational Amplifiers and Linear Integrated Circuits” 4th Edition: PHI.
5. D. Roy Choudhary, Shail Jain “Linear Integrated Circuits”, 4th Edition, New Age International.

Course Code	ECSP3002				
Category	Engineering Science Course				
Course Title	Electronic Devices and Circuits Lab				
Scheme and credits	L	T	P	Credits	Semester
	0	0	2	1	III

List of Experiments

1. PN Junction Diode characteristics and its applications.
2. Design Full wave rectifier with and without Filters and compute its ripple factor.
3. Zener (Avalanche) Diode characteristics and its application.
4. Analyse and verify BJT input & output characteristics.
5. Analyse and verify MOSFET drain & transfer characteristics.
6. Design Inverting operational amplifier with negative feedback and plot its frequency response.
7. Design Non-Inverting operational amplifier with negative feedback and plot its frequency response.
8. Design an Integrator circuit using op-amp IC and analyse its output response for sinusoidal and square wave inputs.
9. Design of Comparator circuit using operational amplifier.
10. Implement Digital to Analog converter (DAC) using operational amplifier.
11. A mini project using discrete electronic components.

Course Code	ECST3003				
Category	Programme Core Course				
Course Title	Digital System Design				
Scheme and credits	L	T	P	Credits	Semester
	3	0	0	3	III

Course Outcomes

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

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

Syllabus

Module I (6 Hrs)

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

Module II (8 Hrs)

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

Module III (6 Hrs)

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

Module IV (6 Hrs)

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

Module V (6 Hrs)

Design of Data Path and Control unit with Case Studies.

Module VI (6 Hrs)

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

Text Book

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

Reference Books

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

Course Code	ECSP 3003				
Category	Programme Core Course				
Course Title	Digital System Design Lab				
Scheme &Credits	L	T	P	Credi ts	Semester
	0	0	2	1	III

List of Experiments

1. Design the following combinational circuits using Verilog HDL. Write a functional simulation test bench to test these designs. Generate all combinations of test inputs with a delay of 5 ns between the set of each test input. Observe the output in the console window.
   a. Full adder using operators
   b. Full adder using two Half adders
   c. 4:1 Mux using 2:1 Mux
   Synthesize the design and count the number of LUT. Plan a pin constraint for any one design for implementation on FPGA.

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

3. Model the 4-bit counter with the following specifications:
   a. Reset synchronous to clock positive edge.
   b. Counter has a 4-bit load terminal to count Up or down from this user-loaded value.
   Write a test bench to test this design and implement it on FPGA.

4. Design a 4-bit sequence detector using Verilog HDL. Write a test bench to test this design.

5. Create a Verilog RAM module with 8-bit wide data and 16 locations, initialize it with data, and perform a read operation to verify data retrieval. Find the average of the retrieved data in the console window.

Course Code	ECST3004				
Category	Engineering Science Course				
Course Title	Discrete Mathematics				
Scheme &Credits	L	T	P	Credits	Semester
	2	0	0	2	III

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

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

Syllabus

Module I (7 Hrs) – Modular Arithmetic
Modular Arithmetic, Euclid's Algorithm, Primes, Fermat's Algorithm, Euler's Theorem, Linear Congruences, Chinese Remainder Theorem, Application to Cryptography.

Module II (7 Hrs) – Graph Theory
Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub-Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Coloring, Coloring maps and Planar Graphs, Perfect Graph.

Module III (7 Hrs) – Lattice Theory
Lattices as partially ordered set, Definitions and Examples, Some properties of Lattices, Lattices as algebraic system, Sub lattices, Direct product, Homomorphism, Some special Lattices.

Module IV (8 Hrs) – Groups and Fields
Group definitions and examples, Cyclic group, Permutation groups, Subgroups and Homomorphism, Co-sets, Lagrange's Theorem, Finite field, Galois field.

Text Books

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

Reference Books

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

Course Code	ECST 3005				
Category	HSSM-VEC				
Course Title	Cyber Laws and Ethics in IT				
Scheme &Credits	L	T	P	Credits	Semester
	2	0	0	2	III

Course Outcomes

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

1. Understand statutory, regulatory, constitutional, and organizational laws for awareness amongst the software professional.
2. Classify Ethics and Laws with respect to legal dilemmas in the Information Technology Act.
3. Illustrate Privacy and Intellectual Property Rights related practices.
4. Categorize business ethics and roles applicable to IT users, IT professional malpractice, IT organization workers.

Syllabus

Module I

Cyber laws and rights in today's digital age; IT Act, Intellectual Property Issues connected with use and management of Digital Data, Emergence of Cyberspace, Cyber Jurisprudence.

Module II

Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Cyber Stalking/Harassment, Cyber terrorism, Cyber Defamation, Different offences under IT Act, 2000, Cyber Torts.

Module III

Ethics in business world, Ethics in IT, Ethics for IT professionals and IT users, IT professional malpractices, Communications eavesdropping, Computer break-ins, Denial-of-service, Destruction and modification of data, Distortion and fabrication of information, Types of Exploits and Perpetrators.

Module IV

Intellectual Property: Copyrights, Patents, Trade Secret Laws, Key Intellectual Property Issues, Plagiarism, Competitive Intelligence, Cyber-squatting, Information Warfare Policy and Ethical Issues.

Module V

Privacy: The Right of Privacy, Protection, Key Privacy and Anonymity Issues, Identity Theft, Consumer Profiling, Defamation, Freedom of Expression, Anonymity, National Security Letters, Defamation and Hate Speech.

Module VI

Ethics of IT Organization: Contingent Workers, H-1B Workers, Whistle-blowing, Protection for Whistle-blowers, Handling Whistle-blowing Situation, Digital Divide.

Text Books

1. George Reynolds, *Ethics in Information Technology*, 5th Edition, Cengage Learning
2. Hon C Graff, *Cryptography and E-Commerce - A Wiley Tech Brief*, Wiley Computer Publisher, 2001

Reference Books

1. Michael Cross, Norris L. Johnson, Tony Piltzecker, *Security*, Shroff Publishers and Distributors Ltd.
2. Debora Johnson, *Computer Ethics*, 3rd Edition, Pearson Education
3. Sara Baase, *A Gift of Fire: Social, Legal and Ethical Issues for Computing and the Internet*, PHI Publications
4. Chris Reed & John Angel, *Computer Law*, OUP, New York, 2007

Course Code	ECST2980				
Category	Open Elective				
Course Title	Basics of Linux Operating System				
Scheme &Credits	L	T	P	Credits	Semester
	2	0	0	2	III

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

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

Syllabus

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

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

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

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

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

Module VI
Remote access and managing processes through remote login.

Text Books

1. *Linux BIBLE*, Christopher Negus, Tenth Edition, Wiley, 2020.
2. *Linux for Beginners: An Introduction to the Linux Operating System and Command Line*, Jason Cannon, O'Reilly, 2014.

Course Code	ECSTH3100				
Course Title	Fundamentals of Artificial Intelligence and Edge computing				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	III

Course Outcomes:

1. Understand the fundamentals of Artificial Intelligence and Edge Computing
2. Apply techniques in edge computing architecture to achieve the best performance of AI training and inference
3. Analyze AI applications on edge under the multiple constraints of networking, communication, computing power, and energy consumption
4. Summarize the principles of problem solving, quantitative and/or qualitative decision making in complex situations on AI
5. Implement edge integration applications

Syllabus

Module I: (5 Hours)
Introduction to Edge Computing: Need, Key Techniques, Benefits, Systems Paradigms of Edge Computing, Edge Computing Frameworks, Value Scenarios for Edge Computing, Edge Computing System Architectures, Industrial Applications of Edge Computing, Intelligent Edge and Edge Intelligence, Challenges and Opportunities in Edge Computing

Module II: (7 Hours)
Paradigms of Edge Computing: Cloudlet and Micro Data Centers, Fog Computing, Mobile and Multi-Access Edge Computing, Edge Computing Terminologies, AI Hardware for Edge Computing, Edge Computing Frameworks, Virtualizing the Edge

Module III: (7 Hours)
AI Applications on Edge: Fundamentals of Artificial Intelligence: Artificial Intelligence and Deep Learning, Neural Networks in Deep Learning, Deep Reinforcement Learning, Distributed DL Training, Potential DL Libraries for Edge. Hybrid Hierarchical Architecture at Three Levels: End, Edge and Cloud; Case Studies of Real-Time Video Analytics, Autonomous Internet of Vehicles, Intelligent Manufacturing, Smart Home and City

Module IV: (7 Hours)
Artificial Intelligence Inference in Edge: Optimization of AI Models in Edge: General Methods, Edge Device, Segmentations of AI Models, Early Exit of Inference (EEoI), Sharing of AI Computation

Module V: (7 Hours)
Artificial Intelligence Training at Edge: Distributed Training at Edge, Federated Learning (FL) at Edge, Communication-Efficient FL, Resource-Optimized FL, Security-Enhanced FL, Case Studies Based on Training at Edge

Module VI: (7 Hours)
Artificial Intelligence Applications on Edge: Real-Time Video Analytics, Autonomous Internet of Vehicles (IoVs), Intelligent Manufacturing, Smart Home and City, Urban Healthcare, Urban Energy Management, Manufacturing, Transportation and Traffic

Text Books:

1. *Edge AI: Convergence of Edge Computing and AI*, Xiaofei Wang, Yiwen Han, Victor C. M. Leung, Dusit Niyato, Xueqiang Yan, Xu Chen

Reference Book:

1. Recent research papers from reputed journals and conferences such as DATE, TEST, CVPR, ICLR, NIPS, ICML, etc.

Course Code	ECSTM3100				
Category	Engineering Science Course				
Course Title	IoT fundamentals				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	III

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

1. Understand the basics of networking
2. Gain the knowledge about IoT standards
3. Realize the basic applications using Arduino and Raspberry Pi
4. Illustrate different real world applications

Syllabus

Module - I: (7 Hrs)
Basics of Networks, TCP/IP model, IP Addresses, application layer protocols, HTTP, MQTT, WWW, constraint application protocol, stacks.

Module - II: (7 Hrs)
Introduction to IoT, evolution of IoT, IoT and SCADA, Big Data, IoT Standards, requirement, Platforms, relevance of IoT, security

Module - III: (7 Hrs)
Interoperability in IoT, Machine-to-Machine Communications, Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Sensing, Actuation, Sensor Networks

Module - IV: (7 Hrs)
Introduction to Python programming, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi.

Module - V: (7 Hrs)
Introduction to SDN, Fog Computing, IoT application case studies: Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT, Agriculture, Healthcare, Activity Monitoring, IoT in India: Smart India projects, Challenges in IoT

Text Books

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

Course Code	ECST4001				
Category	Engineering Core Course				
Course Title	Computer Architecture and Organization				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

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

1. Illustrate common principles of computer organization and multiprocessing
2. Translate the C code into MIPS assembly code
3. Design and test the data and control path of the MIPS processor on FPGA
4. Apply the concept of cache and virtual memory management in computer system
5. Analyse different arithmetic algorithms, control unit and processor datapath with and without pipelining

Syllabus

Module I (4 Hrs)
Introduction to computer system and its sub modules, Introduction to RISC and CISC paradigm, Performance Equation, Common Principles of Computer organization: Amdahl's Law, Principle of Locality.

Module II (5 Hrs)
Processor organization, instruction set (MIPS), instruction formats, Arithmetic for Computers: Addition and Subtraction, Multiplication, Division, IEEE 754 floating point format.

Module III (6 Hrs)
Processor Design-Introduction, Datapath and control unit design, Performance Considerations, Multi-cycle design, Micro Programmed control design, Exception Handling.

Module IV (5 Hrs)
Motivation for Pipelining, Clock period and CPI, Pipelined datapath, graphical representation, Pipelining Hazards.

Module V (5 Hrs)
Memory organization, concepts of semiconductor memory, memory management, concept of cache and associative memories, virtual memory.

Module VI (5 Hrs)
Parallel processing concepts, multiprocessors and its characteristics, Input/Output Subsystem: Interfaces and BUS, I/O Operations, Designing I/O Systems, Case study: Application of RISC and CISC as Data Centers perspective.

Text Books

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

Reference Books

1. Computer Architecture and Organization; J. P. Hayes; Third Edition (Fifth Reprint), McGraw Hill, 2012.
2. Computer Architecture and Parallel Processing; Kai Hawang, Faye A. Briggs, McGraw Hill, 2012
3. Computer Organization; Safwat G. Zaky, Zvonko G. Vranesic, Carl Hammacher; Fifth Edition, McGraw Hill, 2002.
4. Structured Computer Organization; Andrew. S. Tanenbum; Fifth Edition, Pearson, 2005.

Course Code	ECSP4001				
Category	Engineering Core Course				
Course Title	Computer Architecture and Organization Lab				
Scheme &Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

List of Experiments:

1. Integrate ALU and ALU control unit of MIPS processor together and test it using a test bench. Encode the instructions to be executed by ALU and generate these encoded test vectors to control the ALU operations. Supply random test vectors from test bench to check the results using Vivado and FPGA.

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

3. Write a top module to connect the instruction memory, data memory, register files, ALU and ALU control together. Encode one logical and one arithmetic instruction in the instruction memory. Initialize the data memory fields with the data. Verify the ability to fetch instructions from instruction memory. Write a Verilog test bench with arithmetic and logical instruction cases for testing these components.

4. Write a control path in Verilog HDL to control the execution of the data path of MIPS instructions in experiment 3. Analyze the control signals for J type, R type and I type instructions using a suitable test bench.

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

Course Code	ECST4002				
Category	Programme Core Course				
Course Title	Design and Analysis of Algorithms				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

Course Outcomes

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

1. Understand mathematical formulation, complexity analysis and methodologies to solve the recurrence relations for algorithms.
2. Apply Divide and Conquer algorithms and use them in examples
3. Formulate Greedy Methodology and use them in real life examples.
4. Design Dynamic programming and Backtracking Paradigms to solve real-life problems.
5. Design solutions using standard approaches comprehending NP class problems.

Syllabus

Module I

Mathematical foundations for arithmetic and geometric series, Principles of designing algorithms and complexity calculation, Asymptotic notations for analysis of algorithms, worst case and average case analysis, amortized analysis and its applications.

Module II

Divide and Conquer - Introduction to Divide and Conquer, Min Max Problem, Maximum sub-array problem, Closest pair of points problems, Convex hull problem.

Module III

Greedy method – basic strategy, fractional knapsack problem, Minimum cost spanning trees, activity selection problem, find maximum sum possible equal to sum of three stacks.

Module IV

Dynamic Programming - basic strategy, Bellman Ford algorithm, all pairs shortest path, multistage graphs, optimal binary search trees, traveling salesman problem, Longest Common Subsequence problem.

Module V

Basic Traversal and Search Techniques, connected components, Backtracking basic strategy, 8-Queen's problem, sum of subset problem, Introduction to Approximation algorithm.

Module VI

NP-hard and NP-complete - basic concepts, non-deterministic algorithms, NP-hard and NP complete decision and optimization problems, polynomial reduction, vertex cover problem, clique cover problem.

Text Books

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

Reference Books

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

Course Code	ECST 4003				
Category	Programme Core Course				
Course Title	Software Engineering				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

Course Outcomes

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

1. Understand generic processes of software development and learn different techniques and methodologies used in development of software systems.
2. Apply learned concepts to effectively use software testing methodologies in various software development scenarios.
3. Learn to evaluate and develop comprehensive project plans in alignment with project goals and stakeholder needs.
4. Develop comprehensive plans for resource allocation and project monitoring
5. Apply quality management techniques to ensure process and product quality in software development.

Syllabus

Module I

Introduction to Software Engineering, Exploratory style versus Software Engineering, Shortcoming of exploratory style, Basic principles to handle complexity, Some basic issues: Types of software projects, software services, Emergence of software engineering principles, Evolution of design techniques.

Module II

Software Process Models, Basic concepts of classical Waterfall Model, Stages of Waterfall Model, Iterative Waterfall Model, V-Model and Prototyping Model, Incremental Model, Evolutionary Model, Agile Model, Extreme Programming and Scrum, Scrum Life Cycle Model, Case Study on software development life cycle (SDLC)

Module III

Basic testing concepts, levels of testing, Errors, Faults and Failure, Unit, Integration and System Testing, Software Testing fundamentals, Black Box Testing, White Box Testing, Web Testing, Test case design, Path Testing, Case Study on Software Testing Life Cycle (STLC)

Module IV

Software Project management - Plans, Methods and Methodology, The Business Case, Project Success and Failure, Project Evaluation, Cost-benefit evaluation technique, Project Planning - stepwise project Planning, Software Effort Estimation - Albrecht Function Point Analysis, COSMIC Function Point, Cost Estimation, Project Scheduling.

Module V

Resource allocation: Introduction, Nature of Resources, Identifying Resource Requirement, scheduling Resources, Project Monitoring and Control, Project Control Cycle, Configuration Management, Process, Configuration Management Tool, Project Management Tools. Contract Management: Managing Contracts, Project Close out, Project Closure process and report.

Module VI
Software Quality Management: Introduction to Software Quality, Evolution of quality systems, Quality Control, Quality Assurance, Total Quality Management, Process Improvement, Process and Product Quality, CMM (Capability Maturity Model), Personal Software Process (PSP) Software Reliability, Risk management

Text Books

1. Software Engineering-A Practitioner's Approach; Roger Pressman; Sixth Edition, MaGraw Hill, 2010
2. Project Management by Clifford F. Gray, Erik W. Larson, McGraw Hill

Reference Books

1. Software Engineering; Ian Somerville; Seventh Edition; Pearson Education, 2008.
2. Ethics in Information Technology, George W. Reynalds, 4th Edition, Cengage Learning Publication
3. Software Engineering; David Gustafsan, Schaum's Series, Tata McGraw Hill, 2002
4. Software Project Management, Sanjay Mohapatra; First Edition, Cengage Learning, 2011.
5. Software Project Management, Rajib Mall, 5th Edition, McGraw Hill

Course Code	ECSP 4003				
Category	Programme Core Course				
Course Title	Software Engineering Lab				
Scheme &Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

List of Experiments:

1. a. Explore the perspectives and notations of the Unified Modeling Language (UML) in Star UML.
   b. Study the IEEE SRS standard and prepare SRS for the conceptualization of the identified systems.

2. Create a Use-Case diagram to depict the user's perspective of the system, demonstrating user interactions and system functionalities.

3. Develop a Class Diagram to articulate the structural aspects of the system

4. Construct State Diagram to depict the structural view of the system.

5. Construct a Sequence Diagram to represent the dynamic view or behavior of the system, illustrating the chronological flow of interactions among different components or entities within the system.

6. Create Data Flow Diagram to illustrate the system's behavioural perspective.

7. Perform White-Box Testing to test the functionalities using JUnit testing tool.

8. Mini Project: Based on real-time modeling of software on a testbed and in a production environment, with case studies on SDLC and STLC

Course Code	ECST4004				
Category	Engineering Science Course				
Course Title	Statistics for Data Analytics				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

Course Outcomes

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

1. Apply statistical techniques effectively to solve complex problems in the context of data analytics.
2. Apply data cleaning, visualization, and feature engineering for effective exploratory data analysis.
3. Formulate and test hypotheses, construct confidence intervals, and apply statistical inference techniques.
4. Apply statistical inference techniques, queuing theory principles and analyze the structure and behavior of queuing systems.
5. Apply predictive modeling in data analytics through case studies and real-world applications

Syllabus

Module I: Foundations of Statistics (6 Hrs)

Introduction to Descriptive and Inferential Statistics, Measures of Central Tendency and Dispersion, Data Visualization Techniques

Module II: Distributions and Probability Theory (6 Hrs)

Probability theory and Probability Distributions, Sampling and Sampling Distributions

Module III: Exploratory Data Analysis (EDA) (7 Hrs)

Data Cleaning and Preprocessing, Exploratory Data Visualization, Outlier Detection and Handling, Correlation and Covariance Analysis, Feature Engineering Basics (with R Programming)

Module IV: Statistical Inference (7 Hrs)

Hypothesis Testing (Parametric and Non-Parametric), Confidence Intervals, Analysis of Variance (ANOVA), Chi-Square Tests, Power Analysis and Sample Size Determination

Module V: Queuing Theory (6 Hrs)

Structure of a queuing system – Operating characteristics of queuing system – Transient and steady states – Terminology of queuing systems – Arrival and service processes – Pure Birth-Death process Deterministic queuing models – M/M/1 Model of infinite queue – M/M/1 model of finite queue.

Module VI: Advanced Topics in Statistics for Data Analytics (4 Hrs)

Predictive modeling in Data analytics, Case Studies and Real-world Applications

Text Books

1. Gupta S.C. and Kapoor V.K. (2014): Fundamentals of Applied Statistics, Sultan Chand and sons.
2. Agarwal B.L (2007): Basic statistics, 3/e, New Age International (P) Ltd

Reference Books

1. Practical Statistics for Data Scientists – Peter Bruce and Andrew Bruce, O'Reilly Media, Inc.
2. Mood, A. M., Graybill, F. A. and Boes, D.C.: Introduction to the Theory of Statistics, McGraw Hill.
3. "An Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, Springer

Course Code	ECSP4005				
Category	Engineering Core Course				
Course Title	Software Lab-I				
Scheme &Credits	L	T	P	Credits	Semester
	0	0	2	1	IV

Course Outcomes

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

1. Understand Processes, Tools, and Methodologies in Software Development Lifecycle.
2. Implement Agile Software Development Life Cycle.
3. Integrate Software Development and its Operations.
4. Use Cloud Environment and its Services

List of Experiments:

1. Introduction to Dev Ops and SDLC.

2. Git and Git Hub: Creating repository and committing the changes.

3. Executing following tasks using Python FLASK app
   i. Writing and checking endpoints.
   ii. Port Forwarding.
   iii. Adding new endpoint.
   iv. Writing unit tests.
   v. Write a failing unit test.

4. To-do app using FLASK app (CRUD function).

5. Organization and team using GitHub.

6. Docker (Containerization) Part 1:
   i. Create a code space using Flask template.
   ii. Use Linux and Docker commands for different operations
   iii. Read docs from docs.docker.com
   iv. Run commands using nginx
   v. Write and run a “Hello world” python program in Docker.

7. Docker Compose
   i. Fork Repo
   ii. Create Codespaces
   iii. Create and check docker container
   iv. Open Dockerfile and check the contents
   v. Run Docker Compose commands
   vi. Understanding Docker run test suit.

8. Mini Project
   This mini project assessment will primarily focus on your usage of GitHub, with specific emphasis on collaboration and organization, Use Cloud Environment and its Services

Reference Books

1. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations by Gene Kim, Patrick Debois, John Willis, Jez Humble, 2016.
2. Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale by Jennifer Davis.
3. Python for DevOps: Learn Ruthlessly Effective Automation by Noah Gift, Kennedy Behrman, Alfredo Deza, Grig Gheorghiu.
4. Building Microservices: Designing Fine-Grained Systems by Sam Newman.
5. Effective DevOps with AWS: Ship faster, scale better, and deliver incredible productivity by Nathaniel Felsen

Course Code	ECST2990				
Category	Open Elective				
Course Title	Designing with Raspberry Pi				
Scheme &Credits	L	T	P	Credits	Semester
	3	0	0	3	IV

Course Outcomes

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

1. Wire Raspberry Pi and create a fully functional computer
2. Use Python-based IDE and trace and debug Python code on the device
3. Measure physical parameter using sensors
4. Implement various communication protocols for wired and wireless communication
5. Interface different motors and create robots

Syllabus

Module I

Getting started with Raspberry Pi: Basic functionality of the Raspberry Pi and its Processor, setting and configuring the board, differentiating Raspberry Pi from other platforms like Arduino, Beagle, Asus Thinker etc. Overclocking, Component overview.

Module II

Introduction to Linux: Implications of an operating system on the behavior of the Raspberry Pi, Overview of Linux and its terminal command, apt-get update, apt-get upgrade, navigating the file system and managing processes, text-based user interface through the shell, overview of graphic user interface.

Module III

Programming the Raspberry Pi: Python: Introducing to Python programming language; Python Programming Environment, Python Expressions, Strings, Functions, Function Arguments, Lists, List Methods, Control Flow, Numpy, PIP (Python Installation Package) and customized libraries. C++ programming: Basic C++ programming approach, header file structure and library organization, Cross Compiler and its configuration.

Module IV

Exploring Electronics with the Raspberry Pi: Communication facilities on Raspberry Pi (I2C, SPI, UART), working with RPi.GPIO library, Interfacing of Sensors and Actuators.
Project 2: Set up a Pi motion detector
Project 3: Set up a Pi ADC/DAC
Project 4: Construct a digital weather station
Project 5: Construct a Traffic Light Controller

Module V
Communication using Raspberry Pi: Wired and Wireless communication, TCP, IP configurations, SSH, Putty terminal usage.
Project 6: Set up file server
Project 7: Network your keyboard and mouse
Project 8: Create a portable wireless access point
Project 9: Communicate with Arduino
Project 10: Construct a digital server based weather station

Module VI
Robotic Motion PI: DC, Servo, Stepper, Motor Drivers, Motor Shields, Camera Interfacing, remote data logging.
Project 11: Keyboard Control Robot
Project 12: Wireless Robot

Text Books

1. Raspberry Pi 3: An introduction to using with Python Scratch, Javascript and more, Gary Mitnick, CreateSpace Independent Publishing Platform, 2017.
2. Raspberry Pi for Python Programmers Cookbook, Tim Cox, Packt Publishing Limited; 2nd Revised edition, 2016.
3. Raspberry Pi User Guide, Eben Upton and Gareth Halfacree, John Wiley & Sons, 2016.

Course Code	ECST5001				
Category	Programme Core Course				
Course Title	Operating System				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	V

Course Outcomes:

After completion of the course student will be able to:

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

Syllabus:

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

Module II: Process Management, Scheduling Algorithms, Process Synchronization, Threads, and Deadlocks: Process concept and management, Process states and transitions, Process scheduling algorithms, CPU scheduling, Process synchronization and concurrency, Inter-process communication, Deadlock detection and prevention, Resource allocation and management, Multiprogramming and multitasking.**Module III:** Memory Management: Memory hierarchy, Memory allocation strategies, Virtual memory concept, Paging and segmentation, Memory protection and addressing, Memory management unit (MMU).

Module IV: File Systems and I/O Management: File system organization and structure, File system

implementation techniques, File system operations, Disk scheduling algorithms, Device management and drivers, Input/output operations and buffering.

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

Text Books:

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

References:

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

Course Code	ECSP5001				
Category	Programme Core Course				
Course Title	Operating System Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	V

List of Experiments

Experiment 1: Basic OS Commands

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

Tasks:

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

Experiment 2: Shell Scripting Basics

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

Tasks:

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

Experiment 3: Process Management

Objective: Understand process creation, management, and termination.

Tasks:

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

Experiment 4: Inter-process Communication (IPC)

Objective: Explore various IPC mechanisms.

Tasks:

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

Experiment 5: Memory Management

Objective: Understand memory allocation and management.

Tasks:

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

Experiment 6: File Systems and I/O Management

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

Tasks:

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

Experiment 7: Deadlock Detection and Avoidance

Objective: Understand deadlock concepts and implementation.

Tasks:

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

Experiment 8: Scheduling Algorithms

Objective: Explore CPU scheduling algorithms.

Tasks: Implement various scheduling algorithm

Course Code	ECST5002				
Category	Programme Core Course				
Course Title	Embedded System Design				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	V

Course Outcomes

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

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

Syllabus:

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

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

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

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

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

Module VI: **(06 Hrs)**
Structure of µCOS-II: Introduction to µCOS-II-, kernel structure, Task States, Inter task communication, Task Scheduling, Task Synchronization, Critical section, Shared Resources, Context Switching, Priority Inversion, Mutual exclusion. Introduction to embedded Linux.

Text books:

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

Reference Books:

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

Course Code	ECSP5002				
Category	Programme Core Course				
Course Title	Embedded System Design Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	V

Experiment List:

PART-A:

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

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

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

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

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

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

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

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

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

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

PART-B:

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

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

2. Interface and Control a DC Motor.

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

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

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

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

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

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

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

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

Course Code	ECST5003				
Category	Programme Core Course				
Course Title	Digital VLSI Design				
Scheme & Credits	L	T	P	Credits	Semester
	2	1	0	3	V

Course Outcomes:

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

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

Syllabus:

Module I: 5 Hrs

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

Module II: 7 Hrs

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

Module III: 6 Hrs

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

Module IV: 6 Hrs

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

Module V: 6 Hrs
Sequential logic design: Timing Metrics for Sequential Circuits, Classification of Memory Elements, Static Latches and Registers, Dynamic Latches and Registers.

Module VI: 5 Hrs
Clocking Strategies, CMOS Sub-system design: SRAM, DRAM.

Text Book:

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

Reference Books:

1. CMOS VLSI Design: A Circuits and Systems Perspective, Author: N. Weste and K. Eshranghian, Publisher: PHI, Edition: 2nd edition
2. CMOS Digital Integrated Circuits Analysis & Design, Author: S. M. Kang, Yusuf Lablebici, Publisher: TMH, Edition: 3rd edition
3. Basic VLSI Systems and Circuits, Author: Dougles Pucknell and K. Eshraghian, Publisher: PHI, Edition: 3rd edition

Course Code	ECSP5003				
Category	Programme Core Course				
Course Title	Digital VLSI Design Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	V

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

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

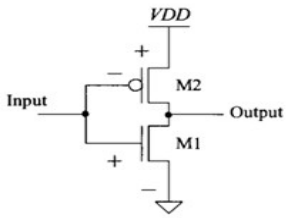


Fig.2

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

4) **Layout and Simulation of an XOR Gate:** Draw the layout of an XOR gate designed for a balanced rise and fall time. Extract the layout, simulate the netlist in SPICE, and measure propagation delay.

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

6) **D Flip-Flop Timing Analysis** Implement D F/F. Define and estimate setup time for your design through SPICE simulation.

7) **Dynamic Logic Design** : Design and simulate a domino logic-based AND-OR-

INVERT (AOI) gate. Analyze its power consumption and speed compared to static CMOS logic.

8) **Effect of Process Variations**: Design and simulate a CMOS inverter in 50nm CMOS technology. Analyze the effect of process variations (e.g., threshold voltage, channel length, and mobility) on its AC and DC characteristics, including transfer curve, propagation delay, and power consumption.

Course Code	ECST5004				
Category	MDM				
Course Title	Machine Learning				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	V

Course Outcomes:

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

1. Perform exploratory data analysis to prepare datasets for machine learning models.
2. Employ calculus, linear algebra, probability theory, and optimization methods to develop machine-learning models.
3. Implement and analyze supervised and unsupervised learning algorithms on a given data set.
4. Evaluate and interpret the performance of machine learning models using evaluation metrics and inference techniques to derive meaningful insights.
5. Apply appropriate machine learning techniques to solve real-world problems by selecting suitable models, algorithms, and optimization strategies.

Syllabus:

Module 1: (6 Hrs)

Foundations for ML: Review of Linear algebra and Optimization, introduction to machine learning and its types, parametric vs non-parametric models, Machine Learning pipeline, Exploratory Data Analysis.

Module 2: (8 Hrs)

Supervised learning algorithms: Linear and Logistic Regression – Bias/Variance Trade-off, overfitting and under fitting, Regularization, Multivariate and polynomial Regression, Variants of Gradient Descent algorithm. Decision Trees, Basic decision trees learning algorithm, Bagging, Boosting, and Random Forests.

Module 3: (8 Hrs)

Support Vector Machines, and Kernel functions in SVM, K-Nearest Neighbors. Feature selection techniques, Feature scaling, Evaluation and Model Selection: Performance Metrics: Accuracy, Precision, Recall, F1-score, Confusion Matrix, ROC & AUC Curves, Evaluation Measures, Cross-Validation techniques.

Module 4: (6 Hrs)

Probabilistic Machine Learning: Bayesian learning and Bayesian networks, Naive Bayes classifier; Bayes optimal classifiers, Maximum Likelihood Estimation, MAP; Gaussian Discriminant Analysis.

Module 5: (7 Hrs)

Unsupervised learning algorithms: K-means clustering, Hierarchical Clustering, DBSCAN, Anomaly Detection: Isolation Forests, One-Class SVM., Dimensionality Reduction techniques: PCA, LDA; Anomaly detection, Recommender System.

Text Book:

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

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

Reference Books:

1. Pattern Recognition and Machine Learning by Christopher M. Bishop, First edition, Springer, 2006.

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

3. Understanding Machine Learning: From Theory to Algorithms by Shai Shale-Shwartz, and Shai Ben-David, Cambridge University Press, 2014.

Course Code	ECSP5004				
Category	MDM				
Course Title	Machine Learning Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	V

List of Experiments:

Lab-01: **Implement data preprocessing techniques on the given dataset.**

a) Perform Exploratory Data Analysis (EDA)
b) Decide on strategies for handling missing data (e.g., imputation, deletion, interpolation).
c) Identify and remove duplicate entries from the dataset if any.
d) Detect outliers and decide on appropriate treatment methods (e.g., removal, transformation, binning).
e) Convert categorical variables into numerical representations suitable for machine learning algorithms (E.g. one-hot encoding, label encoding, and target encoding).
f) Standardize or normalize numerical features to ensure they have a similar scale, preventing certain features from dominating the learning process.
g) Create new features from existing ones or transform existing features to improve model performance (E.g. polynomial features, interaction terms, or domain-specific transformations)
h) Use techniques like filter methods (e.g., correlation analysis), wrapper methods (e.g., recursive feature elimination) for feature selection.
i) Divide the dataset into training and testing sets to evaluate the performance of the machine-learning model.
j) Visualize the dataset to gain insights into its distribution, relationships between features, and potential patterns.
k) Explore summary statistics, histograms, scatter plots, and correlation matrices to understand the data's characteristics and inform preprocessing decisions.

Lab-02: Implement linear regression algorithm (Single, Multiple variable and polynomial) using benchmark datasets and evaluate the performance of linear regression using evaluation measures like MAE, MSE, RMSE, Coefficient of Determination (R^2), and **Adjusted R-squared.**

Lab-03: Implement **the following algorithms to perform the task of classification on the** benchmark datasets and evaluate the performance of algorithms using evaluation measures like Accuracy, Precision, Recall, F1 score, ROC curves, AUC, and cross-validation techniques.

a) Logistic Regression
b) Decision Tree
c) Random Forest

d) K-nearest Neighbor

Lab-04: Build and implement **an image classifier using Support Vector machine (SVM) algorithm** and evaluate the performance of the trained model algorithms using k-fold cross-validation.

Lab-05: Build and develop a model for document classification using probabilistic machine learning algorithms.

Lab-06: Implement the K-means clustering algorithm to perform image segmentation and compare its performance with different numbers of clusters (k) using various evaluation metrics such as silhouette score, Davies-Bouldin index, and within-cluster sum of squares (WCSS).

Lab-07: Perform Dimensionality Reduction using Principal Components Analysis (PCA) and do the following task:

a) Use PCA in order to **visualize** a high-dimensional problem in 2-dimensions.
b) Use PCA in order to **improve model-training time** and understand the **speed-accuracy trade-off.**
c) Evaluate the trade-offs between preserving global structure and local relationships in the data space.
d) Discuss when to use PCA and when not to use it.

Lab-08: Implement **Gaussian Mixture Model (GMM) Clustering** to model complex data distributions and visualize the resulting cluster assignments and probability contours.

Lab-09: Investigate the effectiveness of Isolation Forest for identifying outliers and to detect anomalous behavior in server computers.

Lab-10: Build a simple recommender system using collaborative filtering or matrix factorization techniques and assess its performance using metrics like Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE).

Lab-11: Perform the comparative analysis of ensemble learning techniques on classification tasks.

Lab-12: **A Capstone Project:** Students are required to utilize the knowledge and competencies gained throughout the course to address a practical real-world challenge or investigate a substantial research query within the realm of machine learning.

Course Code	ECST5005-1				
Category	Program Elective Course				
Course Title	Image Processing				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	V

Course Outcomes

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

1. Describe the foundational principles and terminology associated with digital image processing and computer vision.
2. Utilize fundamental image processing algorithms to enhance the visual quality of images, both subjectively and objectively.
3. Extract meaningful features from images for pattern classification and shape analysis.
4. Develop and implement algorithms for solving real life problems in digital image processing.
5. Demonstrate proficiency in implementing algorithms and techniques learned in the course through hands-on projects using tools like OpenCV, MATLAB, or Python libraries.

Syllabus

Module I: (5 Hours)

Fundamentals of Image processing: pixel, domain, size, resolution, and relationship between pixels. Fundamental Steps in Digital Image Processing. Elements of Visual Perception, Photometric image formation, the digital camera, 2D to 3D projections, and Camera Matrix; Motion models: rotation, translation, and affine, scaling, shearing, matrix representation, low rank transforms composition of transformation.

Module II: (8 Hours)

Image Enhancement in Spatial and Frequency Domain: Basic gray level transformations, Histogram equalization, Smoothing Spatial Filters, Order Statistic Filters, Sharpening Spatial Filters. Image smoothing and sharpening using frequency domain filters.

Module III: (7 Hours)

Image Restoration and Denoising: Model of image degradation/restoration processes; Types of image blur, linear position-invariant degradation, estimation of degradation function, linear and nonlinear image restoration techniques, Inverse filtering, Wiener filtering, Constrained Least Squares Filtering.

Image Denoising: Noise Models, restoration in the presence of Noise only, Introduction to latest filtering techniques: bilateral filtering, Non-local mean filter, PCA for image denoising.

Module IV: (8 Hours)

Image Transforms and Compression models: 2D Orthogonal and Unitary Transforms, Discrete Fourier Transform, Discrete Cosine Transform, KL Transform, Wavelets and Multi resolution Processing: Multi resolution Expansions, Wavelet Transforms in 1D and 2D, The Fast Wavelet Transform.

Image Compression: lossless and lossy, JPEG for gray scale and color image compression, Huffman encoding and run length encoding in JPEG, Structure of Huffman encoder and decoder.

Module V: (6 Hours)
Image Segmentation: Detection of Discontinuities, Edge and Corner Detection, Edge linking and Boundary Detection, Hough Transform, Thresholding, Region-Based Segmentation, Graph-based segmentation, Segmentation by clustering, and normalized cuts.

Module VI: (6 Hours)
Feature Extraction and Applications of Image Processing: Boundary and region feature descriptors, SIFT, HOG feature descriptors, Face Recognition, and Image Pattern Classification, Image Compositing and matting.

Text Books

1. Digital Image Processing by R. C. Gonzalez & R. E. Woods, Pearson education, Fourth edition, 2018.
2. Computer Vision: Algorithms and Applications by Richard Szeliski, Springer, second edition, 2022.

Reference Books

1. Anil K. Jain, "Fundamentals of Digital Image Processing," PHI Learning, Indian edition.
2. Digital Image Processing using MATLAB by R. C. Gonzalez, R. E. Woods & Steven Eddins, Pearson education, second edition, 2017.
3. Feature Extraction and Image Processing for Computer Vision by Alberto S. Aguado and Mark S. Nixon, Academic Press, 3 edition, 2012.
4. Image Processing, Analysis and Machine vision by Milan Sonka, Roger Boyle, and Vaclav Hlavac, Cengage India Private Limited, Fourth edition, 2017.
5. Pattern Classification by Richard Duda, Peter Hart, and David Stork, Wiley, Second edition, 2021.

Course Code	ECSP5005-1				
Category	Program Elective Course				
Course Title	Image Processing Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	V

List of Experiments

1. Fundamental operations on an image (such as reading, displaying, rotation, translation, affine, scaling, shearing, image negative).

2. Image enhancement using point processing operations:
   a) Contrast stretching
   b) Gray Level Slicing
   c) Histogram Equalization

3. Image enhancement using global processing operations:
   a) Spatial Filtering (Image smoothing and sharpening)
   b) Frequency Domain Filtering (Image smoothing and sharpening)

4. Image Restoration in the presence of Noise using global processing operations:
   a) Mean Filters
   b) Order-Statistic Filters

5. Image Deblurring using:
   a) Inverse Filtering
   b) Minimum Mean Square Error (Wiener) Filtering

6. Computation of Image Transforms:
   a) Discrete Fourier Transform
   b) Discrete Cosine Transform
   c) KL Transform

7. Implement JPEG compression and decompression model from scratch for a gray scale image.

8. Detection of Edges and Corners in a given image using:
   a) Laplacian operator
   b) Canny edge detector
   c) Marr-Hildreth edge detector
   d) Harris corner detector

9. Performing image segmentation using:
   a) Thresholding
   b) Region Growing
   c) K-means clustering
   d) Graph-Based Segmentation

10. Develop an application for Face Detection using the fundamentals learned in the course.

11. Develop an application for Face Recognition using the fundamentals learned in the course.

12. Develop an application for Texture classification using:
    a) Prototype matching
    b) Optimal statistical formulation

13. A Capstone Project

Course Code	ECST5005-2				
Category	Programme Elective Course				
Course Title	VLSI Signal Processing				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	V

Course Outcomes:

Students will demonstrate the ability to

1. explain the basic understanding of discrete signals and systems
2. apply the pipelining and parallel processing techniques
3. re-design the DSP systems for the given constraints using retiming techniques
4. fold and unfold the DSP systems.
5. analyze data flow in systolic architectures

Syllabus:

Module I: Introduction to Digital Signal Processing (DSP) systems, sampling theorem, discrete time signal & systems, representation of discrete systems in Z domain, basic DSP algorithms and its mathematical representation, representation of DSP algorithms using block diagram, data flow graph, dependence graph, signal flow graph, Loop bound and Iteration bound algorithms, difference between recursive(IIR) and non-recursive (FIR) systems

Module II: Basics of pipelining and parallel processing, cut-set theory, data broadcast structure and transpose structure of FIR systems, fine-grain and course-grain pipelining techniques, realizing a parallel architecture for FIR systems, pipelining and parallel processing for low power.

Module III: Retiming algorithm for IIR systems, cut set retiming and pipelining, retiming for clock period minimization, retiming for register minimization.

Module IV: Unfolding algorithms, properties, unfolding for sample period reduction, word level processing, bit level processing, register minimization techniques using folding transformation.

Module V: Folding algorithms, properties, register minimization techniques using folding transformation.

Module VI: Systolic architecture design, block diagram, space representation, edge mapping table, low level implementations, space time representation for systolic arrays.

Text book:

1) VLSI Digital Signal Processing Systems, Keshab K. Parhi, A Wiley-Interscience Publication, 1999

CourseCode	ECSP5005-2				
Category	Programme Elective Course				
CourseTitle	VLSI Signal Processing lab				
Scheme&Credits	L	T	P	Credits	Semester
	0	0	2	1	V

List of Experiments:

1. Model a recursive and non-recursive DSP filter using Verilog HDL and test its functionality.

2. Apply the cut set theory on 3-Tap FIR filter architecture to design a pipelined FIR system. Demonstrate the increase in the speed of operation. Compare the power results, if the speed of operation is kept same as that of the non pipelined FIR filter.

3. Apply the parallel processing technique on a 3-tap FIR filter architecture to design a 3-parallel FIR system. Demonstrate the increase in the speed of operation.

4. Demonstrate a cut set retiming technique for FIR and IIR filters. Compare the delay and power values.

5. Demonstrate the use of unfolding if the longest node computation time is larger than iteration bound.

6. Demonstrate a folding transformation for DFG of DSP filter architecture.

Course Code	ECST5005-4/ ECSP5005-4				
Category	Programme Elective Course				
Course Title	Cloud Computing				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	2	4	V

Course Outcomes:

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

1. Understand network as recourse for cloud
2. Articulate the concepts of cloud computing
3. Implement the concept of virtualization and resource management.
4. Demonstrate the measures to be taken for handling fault tolerance and security.
5. Provide cloud computing solutions and recommendations for cloud programming and software environments-based applications.

Syllabus:

Module I: Introduction

Introduction to computer network Basics, Computing Services, Servers, Data bases, Networking software, analytics and
intelligence, interconnection of peering points, Autonomous systems.

Module II: Cloud formation

ubiquitous, convenient on demand network access of pooled resource creation, configuration, customization.

Module III: Models of Cloud:

Public, Private and Hybrid Clouds, and service models - Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)methods and technology used.

Module IV: Cloud virtualization: Hardware /Software /Data servers, Networks infrastructure.

Module V: Service providers and their role infrastructure creation/support, security and Administration

Module VI: Use cases and resource Provisioning Oracle cloud infrastructure configuration and management

Text Books:

1. Cloud Computing Principles and Paradigm, Rajkumar Buyya, James Broberg, Andrzej Goscinski, Wiley Publishers.2011

Reference Books:

1. Barrie Sosinsky, - Cloud Computing Bible|| John Wiley & Sons, 2010
2. Tim Mather, Subra Kumaraswamy, and Shahed Latif, - Cloud Security and Privacy An Enterprise Perspective on Risks and Compliance, O'Reilly 2009
3. Cloud Computing: A Practical Approach, Toby Velte, Anthony TVelte, Robert Elsenpeter, McGrawHill,2009
4. Application notes of AWS
5. Application notes of Azure

Course Code	ECSTH5100				
Category	Honor				
Course Title	Image and Video Signal Processing on Edge Processing				
Scheme & Credits	L	T	P	Credits	Semester
	3	1	0	4	V

Course Outcomes:

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

1. Understand the fundamental concepts of digital image processing and various image transforms.
2. Apply image processing techniques in the spatial and frequency domains to enhance and manipulate images.
3. Analyze different image compression models and compare their effectiveness in reducing data size while maintaining quality.
4. Evaluate the steps involved in video processing and assess their impact on video quality and performance.
5. Design and implement efficient image and video processing solutions suitable for edge computing devices.

Syllabus:

UNIT I: Fundamentals of Image Processing and Image Transforms:(7 hrs)
Basics of Digital Image Processing: Image Representation, Sampling, and Quantization, Pixel Relationships and Neighbourhood Operations Image Transforms:2D Discrete Fourier Transform (DFT) and Its Properties, Discrete Cosine Transform (DCT),Hadamard Transform and Its Applications

UNIT II: Image Enhancement and Restoration: (8 hrs)
Spatial Domain Techniques: Histogram Processing, Fundamentals of Spatial Filtering (Smoothing and Sharpening Filters), Frequency Domain Techniques: Basics of Frequency Filtering, Image Smoothing and Sharpening in the Frequency Domain, Image Restoration: Degradation Model, Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration

UNIT III: Image Compression Techniques for Edge Devices: (8 hrs)
Fundamentals of Image Compression: Coding Redundancy, Spatial & Temporal Redundancy, Compression Models: Lossless Compression: Huffman Coding, Arithmetic Coding, Run-Length Coding, Lossy Compression: Transform Coding, Predictive Coding, JPEG Standards. Efficient Compression for Low-Power Edge Devices

UNIT IV: Basics of Video Processing for Edge Devices: (7 hrs)
Analog vs. Digital Video Representation, Time-Varying Image Formation Models, Geometric Image Formation and Sampling of Video Signals, Challenges in Processing Video on Edge Devices

UNIT V: Motion Estimation and Video Compression:(7 hrs)
2D Motion Estimation Techniques: Optical Flow, Pixel-Based and Region-Based Motion Estimation, Ulti-Resolution Motion Estimation, Applications of Motion Estimation in Video Coding, Real-Time Motion Processing on Edge Hardware

UNIT VI: Image and Video Processing on Edge Devices: (8 hrs)
Key Aspects of Edge-Based Image and Video ,Processing Constraints: Computational Power, Memory, Latency, Hardware Considerations: GPUs, TPUs, and AI Accelerators, Common Edge Applications:, Surveillance, Autonomous Vehicles, Healthcare, Smart Cameras, Optimized Image and Video Processing Techniques for Edge: Model Pruning, Quantization, and Approximate Computing, Challenges in Edge-Based Processing: Power Efficiency, Latency, Bandwidth Limitations

Text Book:

1. Digital Image Processing by R. C. Gonzalez & R. E. Woods, Pearson education, Fourth edition, 2018.
2. Computer Vision: Algorithms and Applications by Richard Szeliski, Springer, second edition, 2022.

Reference books:

1. M. Tekalp ,"Digital video Processing", Prentice Hall International Course
2. Image Processing, Analysis and Machine vision by Milan Sonka, Roger Boyle, and Vaclav Hlavac, Cengage India Private Limited,
3. Feature Extraction and Image Processing for Computer Vision by Alberto S. Aguado and Mark S. Nixon, Academic Press,

Course Code	ECSTM5100				
Category	Minor Specialization				
Course Title	Cloud Computing Using Raspberry Pi				
Scheme & Credits	L	T	P	Credits	Semester
	3	1	0	4	V

Course Outcomes:

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

1. Understand the resources and their constricts used in IoT system
2. Understand the programming requirements at different stages
3. Analyse the data steams received at application layer/server
4. Apply the data processing and data acquisition models at controller for serial interface , dashboard, web interface for data aggregation
5. Evaluate the provisioning for resources at servers like analytics, cloud, computing

Syllabus:

Module I: (5 Hrs)
Introduction to Internet of Everything, IoT Reference Model, Different IoT models, Elements in IoT Infrastructure

Module II: (8 Hrs)
IoT Infrastructure Elements and their roles at Different Layer in IoT Reference Model, Devices/ Function of elements in IoT Sensors, Controllers Raspberry-Pi , Network, Cloud, User Applications and Data Analytics

Module III:(7 Hrs)
Introduction to Network layer device and their role as Gateways. Introduction to various access technologies used in IoT

Module IV: (6 Hrs)
Modelling Network communication, Introduction to socket programming, Use cases of different protocols in the networking

Module V: (6 Hrs) Building aCloud Server

Modules VI: (4 Hrs) Use cases of different data gathering paradigms on **different cloud platforms**

Text Book:

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

Reference Books:

1. Internet of Things Principles, Paradigms and Application of IoT, Joseph Kofi Wireko, Kaml Hiran, BPB Publications 2020 1st Edition
2. Cloud-Powered Robotics with Raspberry Pi: Build, deploy, and manage intelligent robots effectively, Edgardo Peregrino, BPB 2023
3. Application Notes of BM280, BF350, HX710B MPS20N0040D
4. Application Notes of PIR, Inductive Capacitive Sensors, IN122 Current Sensor
5. Raspberry Pi application notes

Course Code	ECST3980				
Category	Open Elective				
Course Title	Programming for Vedic Mathematics Sutras				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	V

Course Outcomes:

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

1. Learn to perform basic math operations faster and more easily
2. Develop analytical thinking through Vedic mathematics
3. Use Vedic mathematics to solve real-world problems like temperature conversions, distance conversions, and adding time
4. Study the applications of Vedic Mathematics in number system, fuzzy models, cryptography, VLSI implementation, discrete Fourier transform and digital signal processing.

Syllabus:

Module-I: (08 Hours)

History of Vedic maths, why Vedic maths, salient features of Vedic maths, Vedic maths formulas, 16 sutras, 13 sub sutras, terms and operations. High speed addition by using the concept of computing the whole and from left to right , super fast subtraction by Nikhilam Sutram from basis 100,1000,10,000.

Module-II: (07 Hours)

Multiplication by Urdhavtrighbhyam sutram, multiplication by Vinculum sutram, multiplication by Nikhilam sutram, fast multiplication by 11, multiplication of numbers consisting of all 9s, multiplication of numbers nearest to the base 10, and multiplication of numbers with sub base 50,500,5000.

Module-III: (08 Hours)

Vedic Mathematics in Cryptography; Implementation of RSA Cryptosystem Using Ancient Indian Vedic mathematics and Analysis of cryptographic algorithms based on Vedic mathematics,

Module-IV: (07 Hours)

Vedic Mathematics in Miscellaneous Application: VLSI implementation, Discrete Fourier Transform and digital signal processing

Text books:

1. Sri Bharatikrishna Tirthaji,"Vedic Mathematics",Published by Motilal Banarsidass,

1965.ISBN 81-208-0163-6.

Reference boos:

1. Williams K.R. “Discover Vedic Mathematics” Vedic Mathematics Research Group, 1984.ISBN 1-869932-01-3.

2. Wiliams K.R. and M.Gaskell “The Cosmic Calculator”. Motilal Banarsidass ,2002.ISBN 81-208-1871-7.

3. Nicholas A.P., Williams,J. Pickles.”Vertically and Crosswise”. Inspiration books, 1984.ISBN 1-902517-03

Course Code	ECST6001				
Category	Program Core Course				
Course Title	Database Management System				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

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

1. Apply the basic concepts of Database Systems and Applications
2. Use the basics of SQL and construct queries using SQL in database creation and interaction
3. Design a commercial relational database system (Oracle, MySQL) by writing SQL using the system.

Syllabus:

Module I: (5 Hrs)

Overview of Database systems: Basic concepts Database & Database Users, File System vs. DBMS, Database System Concepts & Architecture, Data models, Schemas & Instances, Structure of Relational Database, The Relational Algebra-Fundamental operators and syntax, Extended Relational Algebra operations

Module II: (7 Hrs)

Entity-Relational Model: Basic concepts, Constraints, Keys, Design Issues, Entity-Relational ship Diagram, Weak Entity Sets, Extended E-R Features, Design of an E-R Features, Transforming ER Model to Relational Data Model.

Module III:(7 Hrs)

Introduction to SQL: Basic Structure, DDL, DML, DCL, structure-creation, alteration, defining constraints- Primary Key, foreign key, unique key, not null, check, IN operator, Set Operations Aggregate Functions, Null Values, Nested Sub-queries, Views, Complex queries, Modification of Database, Joined Relations, Data-Definition Language, Dynamic SQL.

Module IV: (6 Hrs)

Indexing and Hashing: Basic Concepts, Ordered Indices, B+ - Tree Index Files, B-Tree Index Files, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Index Definition in SQL, Multiple-Key Access.

Module V: (6 Hrs)

Transaction Processing: Transactions, ACID Properties, Transaction Schedules & Types, Serializability, Conflict – Serializability, View – Serializability, Testing for Serializability, Concurrency Control: Lock-based Protocols, Time Stamp-based Protocols, Enforcing, Different Locking Modes, 2PL (Two Phase Locking protocol), Multiple Granularity.

Modules VI: (4 Hrs)

Introduction to Advances in Databases: Object-Oriented Databases, Web Databases, Data Warehousing and Mining, Parallel Databases, Distributed Databases.

Text Books:

1. Database System Concepts by Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw Hill Education, 6th Edition, 2013.

2. Database Management Systems by Raghu Ramkrishnanand Johannes Gehrke, McGraw Hill Education, 3rd Edition, 2014.

3. SQL, PL/SQL the Programming Language of Oracle by Ivan Bayross, BPB publications, 4th, 2017.

Reference books:

1. Fundamentals of Database Systems by RamezElmasri and Shamkant B. Navathe, Addison-Wesley, 6th Edition, 2010.

2. PL/SQL Programming: Oracle Programming 11g by Michael McLaughlin, Oracle Press, McGraw Hill Publications, 2013.

Course Code	ECSP6001				
Category	Program Core Course				
Course Title	Database Management System lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	VI

Course Outcomes:

At the end of the course students will demonstrate the ability to ;
1. Apply the basic concepts of Database Systems and Applications
2. Use the basics of SQL and construct queries using SQL in database creation and interaction
3. Design a commercial relational database system (Oracle, MySQL) by writing SQL using the system

Syllabus:

Experiments based on **ECST6001 Syllabus in Database Management System**

Text Books:

1. Database System Concepts by Abraham Silberschatz, Henry F. Korth and S. Sudarshan, McGraw Hill Education, 6th Edition, 2013.

2. Database Management Systems by Raghu Ramkrishnanand Johannes Gehrke, McGraw Hill Education, 3rd Edition, 2014.

3. SQL, PL/SQL the Programming Language of Oracle by Ivan Bayross, BPB publications, 4th Edition, 2017.

Reference books:

1. Fundamentals of Database Systems by RamezElmasri and Shamkant B. Navathe, Addison-Wesley, 6th Edition, 2010.

2. PL/SQL Programming: Oracle Programming 11g by Michael McLaughlin, Oracle Press, McGraw Hill Publications, 2013.

Course Code	ECST6002				
Category	Program Core Course				
Course Title	System Verilog for Verification				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

Upon completion of this course, students should demonstrate the ability to

I. Understand and use the SystemVerilog RTL design and synthesis features, including new data types, literals, procedural blocks, statements, and operators.
II. Apply the SystemVerilog verification features like OOP'S to build classes and construct objects
III. Apply the SV assertions to validate the behavior of the design
IV. Utilize the functional coverage to measure progress during verification, analyze coverage reports and debug coverage holes.

Syllabus:

Module-1: (08 Hrs.)
Verification Guidelines: Introduction, Verification Process, Verification Plan, Verification Methodology Manual, Basic Testbench Functionality, Directed Testing, Methodology Basics, Constrained-Random Stimulus, Functional Coverage, Testbench Components, Layered Testbench,

Module-2: (10 Hrs.)
Data Types and Procedural Statements: Built-in Data Types, Fixed-Size Arrays, Dynamic Arrays, Queues, Creating New Types with typedef, Creating User-Defined Structures, Enumerated Types, Constants, Strings, Procedural Statements, Tasks, Functions, and Void Functions

Module-3: (08 Hrs.)
Basic Object-Oriented Programming: Where to Define a Class, OOP Terminology, Understanding Dynamic Objects

Module-4: (06 Hrs.)
System Verilog Assertions: Types of Assertions and examples

Module-5: (08 Hrs.)
Inter-process Communication and Functional Coverage: Working with Threads, Inter-process Communication, Coverage Types, Functional Coverage Strategies, Simple Functional Coverage Example, Measuring Coverage Statistics During Simulation

Text books:

1. SystemVerilog for Verification: A Guide to Learning the Testbench Language Features, Chris Spear, Springer 2006
2. Writing Testbenches Using SystemVerilog, Janick Bergeron, Springer, 2006

Reference books:

1. Writing Testbenches: Functional Verification of HDL Models, Second edition, Janick Bergeron, Kluwer Academic Publishers, 2003.
2. Open Verification Methodology Cookbook, Mark Glasser, Springer, 2009
3. Principles of Functional Verification, Andreas S. Meyer, Elsevier Science, 2004
4. Assertion-Based Design, 2nd Edition, Harry D. Foster, Adam C. Krolnik, David J. Lacey, Kluwer Academic Publishers, 2004.
5. SystemVerilog for Design: A Guide to Using SystemVerilog for Hardware Design and Modeling, 2nd Edition, Stuart Sutherland, Simon Davidman and Peter Flake, Springer

Course Code	ECSP6002				
Category	Program Core Course				
Course Title	System Verilog for Verification lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	VI

1) Write a self-checking testbench (Monitor) for 16-bit adder. Design a module adder and instantiate it in a testbench. Apply random stimulus to the inputs and display the values of input and output. Extend the testbench code to perform automatic check on addition operation, if addition is not performed correctly for particular set of inputs display the ERROR message.

2) Declare int dynamic array with array size 5. Initialize all the random values between 50 & 100. Display all the values: 1) Display all using %p to
display packed format. 2) Use %d to display all the value use foreach.
3) Using for loop to display.
Declare one more dynamic array of size 5 int data type, assign random value between 10, 20 for each element. Compare both the array using foreach / for loop. Copy first array content to second array using for loop. Compare first array content to second array using for loop.
Resize the first array to 10 elements, while retaining the existing 5 elements. Now print array contents using %p.

3) Declare a Queue of integer data type. Fill Queue with the random values between 50 ,100 Display all the values 1) Display all using %p to display packed format 2) Use %d to display all the value use foreach. Declare one more Queue of size 5 int data type, assign random values between 10, 20. Compare both the queue elements using foreach / for loop. Copy first queue elements to second Queue using for loop. Compare first queue elements to second queue elements using for loop.

4) Define a static function in a module to calculate factorial of 32-bit integer. Define a static variable of integer type and call a function to calculate factorial of numbers from 1 to 7 using for loop. Display the result. Now make the function as an automatic and repeat the same. Conclude your result with the difference between static and automatic methods.

5) Create a class called MemTrans that contains the following members, then construct a MemTrans object in an initial block.
    a. An 8-bit data_in of logic type
    b. A 4-bit address of logic type

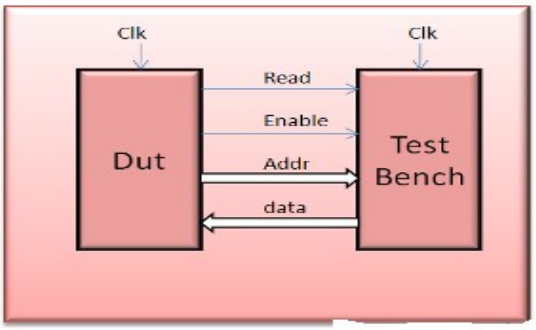
b. A void function called print that prints out the value of data_in and address
Using the MemTrans class, create a custom constructor so that data_in and address are both initialized to 0 but can also be initialized through arguments passed into the constructor. In addition, write a program to perform the following tasks.
    a. Create two new MemTrans objects.
    b. Initialize address to 2 in the first object, passing arguments by name.
    c. Initialize data_in to 3 and address to 4 in the second object, passing arguments by

name.

Modify the solution to perform the following tasks.

d. After construction, set the address of the first object to 4'hF.
    i. Use the print function to print out the values of data_in and address for the two objects.
e. Explicitly deallocate the 2nd object.

6) Write an interface for the following connections between DUT and testbench. Write a module for DUT, where at positive edge of clock if read signal is active then display a message that read is asserted. Write a module for testbench where signal read is initialized to '0' and then after every 20-time unit the read signal is toggled for 3 times. Write a top-level module where clock will get toggled after every 5-time unit and instantiate the interface, DUT and testbench. Consider addr and data is of 8-bit length.



7) Write the SystemVerilog code for the following items.
    a. Create a class Exercise1 containing two random variables, 8-bit data and 4-bit address. Create a constraint block that keeps address to 3 or 4.
    b. In an initial block, construct an Exercise1 object and randomize it. Check the status from randomization.

Modify the solution for above class to create a new class Exercise2 so that:

a. data is always equal to 5
b. The probability of address==0 is 10%
c. The probability of address being between [1:14] is 80%
d. The probability of address==15 is 10%

Using the solution to either Exercise 1 or 2, demonstrate its usage by generating 20 new data and address values and check for success from the constraint solver.
Create a testbench that randomizes the Exercise2 class 1000 times. Count the number of times each address value occurs and print the result.

8) Write a concurrent assertion that states that, at positive edge of clock, if 'cstart' is high that 'req' is high the same clock and 'gnt' is high 2 clocks later. Define a sequence and write property. Assert a property and display the messages for "pass" and "fail" with current time.

Course Code	ECST6003-1				
Category	Programme Elective				
Course Title	Deep Learning-1				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

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

1. Identify key parameters in a neural network's architecture.
2. Develop and implement fully connected deep neural network architecture in a vectorized (efficient) manner.
3. Apply the best practices to train and develop test sets and analyze bias/variance for building deep learning applications.
4. Analyze variance for Deep Learning applications, use standard techniques and optimization algorithms, and check for their convergence.
5. Build and train convolutional neural networks, identify key architecture parameters, and implement vectorized neural networks and deep learning to applications.

Syllabus:

Module 1: (4 Hrs)

Introduction to Deep Learning: Introduction to Neural Networks, Perceptron, McCulloch Pitts Neuron, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Supervised Learning with Neural Networks, Binary Classification using Neural Network, Computation Graph, Vectorization.

Module 2: (6 Hrs)

Shallow Neural Networks: Neural Network Representation, Activation Functions, Feedforward Neural Networks, Representation Power of Feedforward Neural Networks, Gradient Descent for Neural Networks, Backpropagation algorithm.

Module 3: (6 Hrs)

Optimization Algorithms for Deep Learning: Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Eigenvalues and eigenvectors, Eigenvalue Decomposition.

Module 4: (7 Hrs)

Convolutional Neural Networks, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks. Image classification.

Module 5: (5 Hrs)

Autoencoders and relation to PCA, Regularization in Autoencoders, Denoising Autoencoders, Sparse Autoencoders, Contractive Autoencoders.

Module 6: (7Hrs)
Regularization: Bias Variance Tradeoff, L2 regularization, early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout. Greedy Layer wise Pre-training, Better activation functions, Hyper parameters Tuning.

Text Book:

1. Understanding Deep Learning by Simon J. D. Prince, The MIT Press, 2023.
2. Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville, & Francis Bach, MIT Press, 2017.

Reference Books:

1. Deep Learning: A Practitioner's Approach by Adam Gibson and Josh Patterson, Shroff/O'Reilly; First Edition, 2017.

2. Dive into Deep Learning by Aston Zhang, Zachary C. Lipton, Mu Li , Alexander J. Smola, Cambridge University Press, 2024

3. Neural Networks and Deep Learning: A Textbook" by Charu C. Aggarwal, Springer International Publishing, 2018.

4. Deep Learning Illustrated by Jon Krohn, Grant Beyleveld, and Aglaé Bassens, Addison-Wesley; 1st edition, 2019.

5. Recent Research Papers from Reputed Journals and Conferences such as CVPR, ICLR, NIPS, ICML, PAMI etc.

Course Code	ECSP6003-1				
Category	Programme Elective				
Course Title	Deep Learning-1 Lab				
Scheme & Credits	0	T	P	Credits	Semester
	0	0	2	1	VI

List of Experiments:

Lab-01: Implement and use gradient descent (and its variants) with backpropagation for a classification task using a feedforward neural network.

Problem Statement: Implement a feedforward neural network and write the backpropagation code for training the network. We strongly recommend using numpy for all matrix/vector operations. You are not allowed to use any automatic differentiation packages. This network will be trained and tested using the Fashion-MNIST dataset. Specifically, given an input image (28 x 28 = 784 pixels) from the Fashion-MNIST dataset, the network will be trained to classify the image into 1 of 10 classes.

Lab-02: Build and experiment with CNN based image classifiers using a subset of the iNaturalist dataset.

Problem Statement:

b) **Training from Scratch**: Build a small CNN model consisting of 555 convolution layers. An activation and a max-pooling layer would follow each convolution layer. After 555 such conv-activation-maxpool blocks, you should have one dense layer followed by the output layer containing 101010 neurons (111 for each of the 101010 classes). The input layer should be compatible with the images in the iNaturalist dataset. The code should be flexible such that the number of filters, size of filters, and activation function of the convolution layers and dense layers can be changed. You should also be able to change the number of neurons in the dense layer.

c) **Fine-tuning a pre-trained model**: In most DL applications, instead of training a model from scratch, you would use a model pre-trained on a similar/related task/dataset. From `torchvision`, you can load **ANY ONE** model (`GoogLeNet`, `InceptionV3`, `ResNet50`, `VGG`, `EfficientNetV2`, `VisionTransformer` etc.) pre-trained on the ImageNet dataset. Given that ImageNet also contains many animal images, it stands to reason that using a model pre-trained on ImageNet maybe helpful for this task. You will load a pre-trained model and then fine-tune it using the naturalist data that you used in the previous question. Simply put, instead of randomly initializing the weights of a network you will use the weights resulting from training the model on the ImageNet data (`torchvision` directly provides these weights). Please answer the following questions:

Lab-03: Build and experiment with an Auto encoder model for performing the following task.

b) Image Denoising
c) Image Compression

Lab-04: Build, train and evaluate Sparse Autoencoders and Contractive Autoencoders for performing the following task.

b) **Domain Adaptation**
c) Anomaly Detection

Lab-05: Regularization and Hyper parameter Tuning

b) Analyze the bias-variance tradeoff and implementing L2 regularization in neural networks.
c) Experiment with early stopping and dataset augmentation techniques.
d) Implementing dropout regularization and comparing its effectiveness with other techniques.
e) Explore hyper parameter tuning strategies for optimizing neural network performance.

Lab-06: Deployment of deep learning model:

a) Deploy the pertained model directly on edge devices (e.g., smartphones, IoT devices) to assess real-time performance and resource constraints.

b) Deploy the model on cloud platforms (e.g., AWS, Google Cloud, Azure) using services like AWS Lambda, Google Cloud Functions, or Kubernetes clusters. Evaluate scalability, cost, and ease of deployment.

c) Use Docker or similar containerization tools to encapsulate the model and its dependencies. Compare deployment and runtime performance with non-containerized approaches.

Lab-07: Implementing and Presenting a Research Paper Algorithm from Prestigious Conferences (CVPR, ICCV, BCCV, and ECCV) During Lab Sessions.

Course Code	ECST6003-2				
Category	Programme Elective Course				
Course Title	C Based VLSI Design				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

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

1. Understand the High-Level Synthesis (HLS) process and its role in converting C-code to hardware.
2. Apply scheduling, allocation, and binding techniques for efficient hardware synthesis.
3. Optimize C-code for improved hardware performance using loop transformations and dataflow optimizations.
4. Implement verification techniques to ensure correctness and equivalence between C and RTL.
5. Explore advanced topics like HLS for security, FPGA technology mapping, and recent trends in VLSI design.

Syllabus:

Module I:

Introduction to High-Level Synthesis (HLS) (5 Hours)

Introduction to High-Level Synthesis (HLS), fundamentals of HLS design flow, scheduling, resource allocation, binding, and controller synthesis, introduction to scheduling techniques, **ILP formulation for MLRC and MRLC scheduling**, introduction to multiprocessor scheduling and its algorithms, list-based scheduling for MLRC and MRLC, Forced Directed Scheduling, Forced Directed MLRC and MRLC Scheduling Algorithm, Path-Based Scheduling techniques, role of constraints in HLS, and performance trade-offs in scheduling.

Module II:

Allocation and Binding Techniques in HLS (5 Hours)

Understanding the **allocation and binding problem formulation,** Left Edge Algorithm for binding**, ILP formulation of allocation and binding**, register allocation and binding for efficient design, hierarchical graph-based allocation and binding, multi-port binding problem and its impact on resource sharing, impact of clock constraints on binding**, datapath and controller synthesis** techniques, interconnect synthesis and trade-offs in datapath architecture.

Module III: HLS for Arrays and Loops (5 Hours)

High-Level Synthesis for array-based designs, handling memory in HLS, **HLS for loops**, impact of loop transformations on hardware, **HLS for loop pipelining,** loop unrolling, loop fusion, software pipelining, initiation interval minimization, design trade-offs in loop optimizations, scheduling loops for minimum latency and area, impact of memory hierarchy on loop optimizations, loop tiling and array partitioning, and array dependency analysis in HLS.

Module IV:

Optimizing C-Code for Hardware Efficiency (5 Hours)

Hardware-efficient C coding principles, writing C-code with HLS-aware optimizations, **dataflow optimization in HLS**, control and data dependency analysis, **frontend optimizations in C**, handling function calls and recursion in HLS, effects of compiler optimizations on circuit performance, hardware/software partitioning, area-power-performance trade-offs in C-based HLS, resource sharing and function inlining, impact of precision tuning and bit-width reduction in C-based hardware design.

Module V:

Verification and Reverse Engineering in HLS (5 Hours)

Simulation-based verification strategies for HLS-generated designs, phase-wise verification of HLS flow, verification of synthesized RTL, testbench creation for HLS-based designs, equivalence between C and RTL, debugging and refining synthesized designs, validation challenges in C-to-RTL transformation, RTL to C reverse engineering, synthesis validation methodologies, formal verification for HLS, and integrating HLS verification into existing ASIC/FPGA flows.

Module VI:

Advanced Topics in C-Based VLSI Design (5 Hours)

Introduction to hardware security, attacks on RTL logic locking and countermeasures, introduction to logic synthesis, FPGA technology mapping and optimizations, introduction to physical synthesis, circuit-level optimizations for power and performance, recent advances in C-based VLSI design, emerging trends in HLS-based ASIC and FPGA design, case studies on commercial HLS tools and research directions in high-level synthesis.

Text Book:

1. Mike Fingeroff, High-Level Synthesis Blue Book, Mentor Graphics Corporation, 2010.

2. D. D. Gajski, N. D. Dutt, A.C.-H. Wu and S.Y.-L. Lin, High-Level Synthesis: Introduction to Chip and System Design, Springer, 1st edition, 1992

Reference Books:

1. G. De Micheli. Synthesis and optimization of digital circuits, McGraw Hill, India Edition, 2003.

2. Philippe Coussy and Adam Morawiec, High-level Synthesis from Algorithm to Digital Circuit, Springer, 2008

Course Code	ECSP6003-2				
Category	Programme Elective Course				
Course Title	C Based VLSI Design Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	VI

List of Experiments:

1. Basic HLS Flow: C to RTL Conversion: To understand the fundamental workflow of **High-Level Synthesis (HLS)** by converting a simple **C function into RTL**, analyzing resource utilization, and exploring the impact of coding styles on synthesized hardware.

2. Scheduling Techniques in HLS: To explore **various scheduling techniques** such as **list-based scheduling, force-directed scheduling, and ILP-based scheduling**, and analyze their impact on computation latency, parallelism, and hardware efficiency.

3. Allocation and Binding Optimization: To study **allocation and binding techniques** in HLS, focusing on **register allocation, multi-port binding, and hierarchical graph-based binding**, and optimize resource sharing for efficient datapath synthesis.

4. Hardware-Efficient C Coding & Dataflow Optimization: To examine the impact of **C coding styles on hardware efficiency**, implement **dataflow optimizations**, and apply **memory partitioning, function in lining, and loop transformations** to enhance performance.

5. Verification and Equivalence Checking in HLS: To perform **C-to-RTL verification**, apply **simulation-based testing**, check **equivalence between C and RTL**, and debug mismatches using waveform analysis in Vivado HLS.

6. Advanced HLS: FPGA Implementation & Security: To implement **HLS-based designs on FPGA**, explore **technology mapping and logic synthesis**, and study security vulnerabilities such as **hardware Trojans and logic obfuscation** in synthesized RTL.

Course Code	ECST6003-4/ECSP6003-4				
Category	Micro Specialization General Program Elective-II				
Course Title	Data mining and Warehousing				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	2	4	VI

Course Outcomes

1. Understand the Database models IoT reference Model and different element used in it.
2. Apply the understanding in identifying the database system environment
3. Analyse the role of different knowledge discovery databases (KDD)
4. Propose a solution to a real-world problem Backend-frontend system
5. Evaluate the engineering feasibility of the solutions' based on the data storage and retrieval methods

Syllabus:

Module 1:

Introduction to database system Environment, Different models of Databases

Module 2:

Data storage system, Datamining knowledge discovery Databases(KDD), prediction of data in databases, different models of training

Module 3:

Data warehousing component: Database(Backend), Data source(UI), Data loaders(Engines), query and analysis tools(Front End)

Module 4:

Approaches of schema design, data transformation – cleansing, analytical Progressing (Aggregation)

Module 5:

Use cases of different data mining algorithm's, Data warehousing in Oracle cloud infrastructure

Text Book:

1. Data Mining and Data Warehousing: Principles and Practical Techniques, Pratik Bhatia, Cambridge Press, 2019 1st Edition

Reference Books:

1. Oracle Data Mining Concepts
2. Oracle Data Mining User's Guide
3. Oracle Database-Data warehousing Guide

4. Oracle Cloud Infrastructure Application Note
5. Oracle Live SQL Server Application Note

Practical's: The hands-on lab will be based on the following

1. Understand the Database System environment and database models
2. Identification of data storage sources and
3. Analyse the role and functions of different KDD in prediction of data
4. Proposing a solution based on different training models for data mining
5. Evaluating the feasibility of the data mining and warehousing system on the parameters
    a. Hardware platform/ resource usage
    b. Latency
    c. Downtime if any requirements
    d. Scale and volume of data
    e. Security and Maturity

Course Code	ECST6004-1				
Category	Programme Elective Course				
Course Title	Natural Language Processing				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes

1. Understand the fundamental tasks in natural language processing (NLP) related to syntax, semantics, and pragmatics.
2. Apply knowledge of natural language annotation techniques and text analysis tools.
3. Understand and apply statistical parsing techniques.
4. Implement Semantic Role Labeling (SRL) and Semantic Parsing techniques for understanding sentence meaning and Information Extraction.
5. Identify issues and challenges in Machine Translation.

Syllabus

Module I: Introduction (6 Hours)
NLP tasks in syntax, semantics, and pragmatics. Key issues & Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. Brief history of the field.

Module II: N-gram Language Models (6 Hours)
Role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models. Part of Speech Tagging and Sequence Labeling Lexical syntax. Hidden Markov Models. Maximum Entropy models.

Module III: Syntactic Parsing (6 Hours)
Grammar formalisms and tree banks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs.

Module IV: Semantic Analysis (7 Hours)
Lexical semantics and word-sense disambiguation. Compositional semantics. Semantic Role Labeling and Semantic Parsing.

Module V: Information Extraction (IE) (5 Hours)
Named entity recognition and relation extraction. IE using sequence labeling. Automatic summarization. Subjectivity and sentiment analysis.

Module VI: Machine Translation (MT) (5 Hours)
Basic issues in MT. Statistical translation, word alignment, phrase-based translation, and synchronous grammars.

Textbook

1. D. Jurafsky and R. Martin, *Speech and Language Processing*, 2nd edition, Pearson Education, 2009.
2. Allen and James, *Natural Language Understanding*, Second Edition, Benjamin/Cumming, 1995.
3. Charniack & Eugene, *Statistical Language Learning*, MIT Press, 1993.

Reference Book

1. Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal, *NLP: A Paninian Perspective*, Prentice Hall, New Delhi, 1994.
2. T. Winograd, *Language as a Cognitive Process*, Addison-Wesley, 1983.

Course Code	ECST6004-2				
Category	Programme Elective Course				
Course Title	Design for Testability				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

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

1. Describe the Testability of Combinational Circuits
2. Illustrate the concepts of Built In Self-Test
3. Demonstrate the design for Testability of Memory Circuits
4. Illustrate Self Checking Circuits using various techniques

Syllabus:

Module 1: (05 Hrs)
Design for Testability for Combinational Circuits:
Stuck at Faults, Fault diagnosis by Path Sensitization Technique, Reed Muller's expansion technique, OR-AND-OR design, Automatic Synthesis of Testable Logic, Testable design of Multilevel Combinational Circuits

Module 2: (7 Hrs)
Design for Testability for Sequential Circuits:
Controllability and observability, Ad-Hoc Design Rules for Improving Testability, Scan Path Technique for testable Sequential Circuit design, Level Sensitive Scan Design (LSSD), Random Access Scan Technique, partial Scan, Boundary Scan.

Module 3: (06 Hrs)
Built-In Self-Test:
Test Pattern generation for BIST, Output Response Analysis, Circular BIST, Built-In logic Block observer, Self-Testing using an MISR and Parallel Shift register Sequence generator, LSSD On-Chip Self-Test.

Module 4: (06 Hrs)
Testable Memory Design:
RAM fault Models, Test Algorithms for RAMs-Galloping 0's and 1's,Walking 0's and 1's, March Test, MATS Check Board Test, Detection of Pattern-Sensitive Faults, BIST Techniques for RAM Chips.

Module 5: (05 Hrs)
Self - Checking Circuits:
Basic concepts of Self checking circuits, Design of Totally Self Checking checker- Self Checking using m/n codes, Equality Checkers, Berger code, Self-Checking Combinational Circuits, Self - Checking Sequential Circuit.

Text Book:

1. Lala, Parag K. An Introduction to Logic Circuit Testing, Morgan & Claypool, 2009.
2. Parag K. Lala, Digital Circuits Testing and Testability, Academic Press, 1997.
3. M. Abramovili, M.A. Breues, A. D. Friedman, Digital Systems Testing and Testable Design, Jaico publications, 2001.

Reference Books:

1. Zainalabedin Navabi, Digital System Test and Testable Design Using HDL Models and Architectures, Springer, 2011.
2. Parag K. Lala, Fault Tolerant & Fault Testable Hardware Design, PS Publications, 2002.
3. Weste and Eshraghian, Principles of CMOS VLSI Design, Pearson Education, 2nd edition,2000.

Course Code	ECSP6004-2				
Category	Programme Elective Course				
Course Title	Design for Testability Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	VI

Course Outcomes:

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

1. Describe the Testability of Combinational Circuits
2. Explain the Testability of Sequential Circuits
3. Illustrate the concepts of Built In Self-Test
4. Demonstrate the design for Testability of Memory Circuits
5. Illustrate Self Checking Circuits using various techniques

List of experiments of design for testability:

1. Implementation of combinational circuit and apply the path sensitization technique to detect stuck-at faults.
2. Convert a given Boolean function into Reed Muller's canonical form and analyze its testability.
3. Modify a given sequential circuit to include a scan path and analyze the improvement in testability
4. Implement a random-access scan technique in a sequential circuit and verify its functionality.
5. Apply boundary scan techniques to test a sequential circuit for structural faults.
6. Design a circuit with a built-in self-test module, generate test patterns, and analyze responses.
7. Implement an MISR and a parallel shift register sequence generator to perform self-testing of a circuit.
8. Simulate a RAM module and apply the March Test to detect memory faults.
9. Implement a parity checker with Berger codes and analyze its fault detection capability.
10. Design a combinational circuit and apply testability rules to generate an automatically testable version.

Course Code	ECST6004-4				
Category	Programme Elective Course				
Course Title	Big Data Web Intelligence				
Scheme & Credits	L	T	P	Credits	Semester
	3	0	0	3	VI

Course Outcomes:

Students will be able to:

1. Understand Big Data concepts and distributed processing frameworks.
2. Design cloud-based Big Data storage and data pipelines.
3. Extract insights from web and social media data.
4. Develop intelligent web systems using AI and ML techniques.
5. Visualize data and evaluate real-world Big Data solutions using Oracle tools

Syllabus:

Unit I: Foundations of Big Data & Distributed Systems (7 Hours)

Introduction to Big Data, Distributed File Systems & Processing, Big Data Ecosystem, Apache Spark Ecosystem

Unit II: Big Data Storage, Cloud Systems & Data Engineering (7 Hours)

Modern Big Data Architectures, Cloud Infrastructure for Big Data, Data Engineering & Pipelines, Big Data Analytics

Unit III: Web Intelligence Foundations (7 Hours)

Introduction to Web Intelligence, Web Mining, Web Data Extraction, Social Media Intelligence

Unit IV: Intelligent Systems for the Web (7 Hours)

Recommender Systems, Search & Information Retrieval, Semantic Web & Knowledge Graphs, AI & Machine Learning for Web Data

Unit V: Visualization, Ethics & Case Studies (7 Hours)

Data Visualization & Reporting, Privacy, Security & Governance, Industry Case Studies, Capstone Project (Oracle Platform)

Text Books:

1. Hadoop: The Definitive Guide, Fourth Edition by Tom White, O'Reilly Media 2015

Reference Books:

1. Big Data: Concepts, Technology, and Architecture, Balamurugan Balusamy, R Nandhini Abirami, Seifedine Kadry, Amir H. Gandomi
2. Web Intelligence, Ning Zhong, Jiming Liu, Yiyu Yao, Springer Berlin, Heidelberg
3. Hacking Web Intelligence by Sudhanshu Chauhan, Nutan Kumar Panda, O'Reilly Media, 2015

Course Code	ECSP6004-4				
Category	Programme Elective Course				
Course Title	Big Data Web Intelligence Lab				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	2	1	VI

Experiments will be based on:

1. Install.
2. configure and run Database infrastructure.
3. Installation of Hadoop.
4. Data Analysis & Visualization: Exploratory Data Analysis (EDA).
5. Data Visualization Project.
6. Simple Search Engine .
7. Pricing Prediction etc.

Course Code	ECST6005				
Category	MDM				
Course Title	Data Handling and Visualization				
Scheme & Credits	L	T	P	Credits	Semester
	2	0	0	2	VI

Course Outcomes:

After completion of the course student will be able to:

1. **Understand and apply data handling techniques** – Perform data collection, cleaning, and preprocessing to manage structured and unstructured data efficiently.
2. **Demonstrate proficiency in basic data visualization** – Utilize tools like Matplotlib and Seaborn to create fundamental plots such as bar charts, histograms, and scatter plots.
3. **Implement advanced visualization techniques** – Apply multivariate, time-series, and distribution-based visualizations to analyze complex datasets effectively.
4. **Develop interactive dashboards and reports** – Use tools like Power BI or Tableau to create visual analytics for data-driven decision-making.
5. **Apply geospatial and specialized visualizations** – Implement maps, network graphs, and hierarchical plots to analyze spatial and relational data.
6. **Evaluate ethical considerations in data visualization** – Identify misleading visualizations and apply best practices to ensure accurate and ethical data representation.

Syllabus:

Module I: Introduction to Data Handling
Understanding Data: Types, Sources, and Structures, Data Collection, Cleaning, and Preprocessing, Handling Missing Data and Outliers, Data Wrangling Techniques using Pandas and NumPy

Module II: Fundamentals of Data Visualization
Importance of Data Visualization, Basic Principles of Effective Visualization, Introduction to Matplotlib and Seaborn, Creating Line, Bar, Scatter, and Histogram Plots

Module III: Advanced Visualization Techniques
Multivariate Data Visualization, Heatmaps, Boxplots, and Violin Plots, Time Series Visualization, Interactive Visualizations using Plotly and Bokeh

Module IV: Data Visualization for Decision Making
Storytelling with Data, Dashboards and Reports, Data Aggregation and Summary Statistics, Business Intelligence Tools (Power BI, Tableau)

Module V: Geospatial and Specialized Visualizations
Introduction to Geospatial Data, Choropleth Maps and Spatial Data Visualization, Network Graphs and Tree Maps, Visualization in Big Data Analytics

Module VI: Real-World Applications and Case Studies
Industry-Specific Visualization Techniques (Healthcare, Finance, etc.), Best Practices and Common Pitfalls in Data Visualization, Data Ethics and Misleading Visuals, Final Project: Analyzing and Visualizing a Real Dataset

Text Books:

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

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

References:

1. Nelli, F., (2018), Python Data Analytics With Pandas, NumPy, and Matplotlib Links to an external site., (2nd ed.), Springer, New York.

2. VanderPlas, J., (2016), Python Data Science Handbook: Essential Tools for Working with Data Links to an external site.(1st ed.), Sebastapol CA, O'Reilly Media, Inc.

Course Code	ECSP6006				
Category	Vocational and Skill Enhancement Course				
Course Title	Software Laboratory-II (Mobile App Development)				
Scheme & Credits	L	T	P	Credits	Semester
	0	0	4	2	VI

Course Objective:

The primary objective of this course is to equip students with hands-on experience in designing, developing, testing, and deploying mobile applications for Android and iOS platforms. The course covers both native and cross-platform mobile app development, focusing on user experience, performance, and security.

Course Outcomes (COs):

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

CO1: Understand the fundamentals of mobile app development, including Android architecture, UI design, and activity lifecycle.

CO2: Develop interactive user interfaces and implement event-driven programming in mobile applications.

CO3: Utilize data storage techniques such as SharedPreferences, SQLite, and Firebase for managing app data.

CO4: Integrate REST APIs, device sensors, and location services to enhance app functionality.

CO5: Design and develop a complete mobile application incorporating UI/UX, database, and networking features.

Syllabus: List of Experiments:

1. **Installation and Setup of Android Studio for Mobile App Development**
    a. Installing Android Studio, SDK, and setting up an emulator.
    b. Running a "Hello World" Android app.

2. **Designing User Interfaces Using XML and Implementing Event Handlers in Android**
    a. Creating UI elements like Buttons, TextViews, and EditTexts.
    b. Handling user interactions through event listeners.

3. **Understanding Activity Lifecycle and Implementing Intents for Navigation in Android Apps**
    a. Exploring Android activity lifecycle and state management.
    b. Implementing explicit and implicit intents for navigation between activities.

4. **Developing Dynamic Lists Using RecyclerView and ListView in Android Applications**
    a. Creating lists with RecyclerView/ListView.
    b. Handling user interaction with list items.

5. **Creating User Input Forms with Validation in Android Applications**
    a. Building login and registration forms.
    b. Implementing form validation techniques.

6. **Implementing Data Storage Using SharedPreferences and SQLite in Android Apps**
    a. Using SharedPreferences for storing small data like user settings.
    b. Implementing SQLite for structured data storage.

7. **Fetching and Displaying Data from REST APIs Using Retrofit or Volley in Android**
    a. Making network requests to fetch real-time data.
    b. Parsing JSON responses and displaying data in UI.

8. **Accessing Device Sensors and Implementing Google Maps for Location Services**
    a. Using accelerometer and gyroscope sensors in an app.
    b. Integrating Google Maps API for location tracking.

9. **Integrating Multimedia and Push Notifications Using Firebase Cloud Messaging (FCM)**
    a. Playing audio and video files within the app.
    b. Implementing Firebase Cloud Messaging (FCM) for push notifications.

10. **Development of a Mini Project Incorporating Multiple Features of Android App Development**
    a. Developing a complete mobile application that integrates UI, database, networking, and sensors.
    b. Example: To-do List App, Weather App, Notes App, or any other functional app.

Course Code	ECSTH6100				
Category	HONORS Specialization				
Course Title	Computer vision with Embedded Machine Learning				
Scheme & Credits	L	T	P	Credits	Semester
	3	1	0	4	VI

Course Outcomes:

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

1. **understand and Apply Basic Concepts of Computer Vision**
2. **implement and Evaluate Machine Learning Models for Computer Vision Tasks**
3. **integrate Machine Learning Models into Embedded Systems**
4. **deploy machine learning models on hardware for computer vision applications**
5. integrate computer vision techniques and embedded machine learning, demonstrating practical applications

Syllabus:

Module-I (10 Hrs):

Introduction to Computer Vision, Key applications and use cases, Scope and significance in modern technology, Image Representation and Properties**,** Pixel, image resolution, and color spaces, Image formats and their differences, Image Transformation and Filtering, Convolution and correlation, Smoothing and sharpening filters, Edge Detection and Segmentation, Feature Extraction

Module-II (12 Hrs):

Convolution operation and kernels, Layer types: convolutional, Pooling layers and their significance, fully connected, dropout, Data augmentation techniques, Common architectures: CNN, Inception V1/V2/V3, Introduction to AdaNet, Transfer learning and fine-tuning pre-trained models, Regularization techniques to prevent overfitting.

Module-III (10Hrs):

Overview of Embedded Systems for Computer Vision, **Hardware Platforms for Computer Vision,** Camera modules and interfacing techniques, **Real-Time Image Processing,** Techniques for real-time processing: buffering, pipelining, Optimization for speed and efficiency on embedded devices, Configuring Raspberry Pi/NVIDIA Jetson for image capture, Implementing simple image processing tasks on embedded hardware,

Module-IV (08 Hrs)

Overview of object detection: key concepts and terminology, Traditional Object Detection Methods, Modern Object Detection Techniques, Deploying object detection models on embedded hardware,

Text Books

1. "Deep Learning for Computer Vision", Rajalingappaa Shanmugamani, Packt Publishing, 2018.
2. Computer Vision: Algorithms and Applications by Richard Szeliski, Springer, second edition, 2022.

Reference Book:

- Practical Deep Learning for Cloud, Mobile, and Edge by Anirudh Koul, Siddha Ganju, and MeherKasam, O'Reilly Media, 1st Edition: 1st , 2019.

Course Code	ECSTM6100				
Category	MINOR Specialization				
Course Title	Data Management and Analytics for IoT				
Scheme & Credits	L	T	P	Credits	Semester
	3	1	0	4	VI

Course Outcomes:

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

1. Assess various storage and retrieval methods through appropriate indexing design
2. Analyze efficiency of algorithms for database operations comprehend contemporary database architectures and its relevant issues
3. Evaluate the role of database management systems in IOT applications.
4. Design and implement properly structured databases that match the standards based under realistic constraints and conditions.

Syllabus:

Module 1: (05 Hrs)

Data storage: Overview of RDBMS concepts, Basic File Structures, File Organization & Record formats, Heap sorted & Hashed Files, Buffer management, Disk Storage, Parallel Disk access with RAID, Modern Storage Architectures

Module 2: (07 Hrs)

Indexing Structures: Single level and Multilevel Indexes, B Tree and B+ Tree Indexes, Hash and bitmap based indexing, Index Structures for Single Dimensional and Multidimensional Databases

Module 3: (08 Hrs)

Query Processing: Query Execution, Algebra for Queries, Physical-Query-Plan-Operators, Algorithms for Database Operations, Algorithms for Joins and Sorting, hash and index based algorithms, Buffer Management, Parallel Algorithms for Relational Operators

Module 4: (07 Hrs)

Query Optimization: Algebraic Foundation for Improving Query Plans, Estimating Cost of Operations, Cost Based Plan Selection, Choosing Order of Joins, Optimization of Queries for Parallel, Distributed, Multidimensional and Text Database

Module 5: (06 Hrs)

Sustainability Data and Analytics in Cloud-Based M2M Systems - potential stakeholders and their complex relationships to data and analytics applications – Social Networking Analysis - Building a useful understanding of a social network – Leveraging Social Media and IoT to Bootstrap Smart Environments : lightweight Cyber Physical Social Systems – citizen actuation

Module 6: (06Hrs)

Apache Hadoop, Using Hadoop MapReduce for Batch Data Analysis, Apache Oozie, Apache Spark,

Apache Storm, Using Apache Storm for Real-time Data Analysis.

Text Books:

1. Stackowiak, R., Licht, A., Mantha, V., Nagode, L.," Big Data and The Internet of Things Enterprise Information Architecture for A New Age", Apress, 2015.
2. Dr. John Bates, "Thingalytics - Smart Big Data Analytics for the Internet of Things", john Bates, 2015.

Reference Books:

1. RamezElmasri, Shamkant B Navathe, Fundamentals of Database System, Pearson Education
2. Garcia Molina, Ullman, Widom, Data Base System Implementation, Pearson education
3. Raghu Ramakrishnan& Johannes Gehrke, Database Management Systems, McGraw Hill
4. Silberschatz, Korth, Sudarshan, Database System Concepts, McGraw Hill
5. M.TamerOzsu, Patrick Valduriez, S.Sridhar, Principles of Distributed Database Systems, Pearson Education