BVVS

Basaveshwar Engineering College, Bagalkote 587102
Department of Electronics and Computer Engineering

Scheme and Syllabus for B. E.

in

Electronics and Computer Engineering

(for 2025-26 Admitted Batch)

BVVS

Basaveshwar Engineering College, Bagalkote 587102
Department of Electronics and Computer Engineering

Vision of the Institute

To be an institution of excellence in education, research and innovation for a sustainable future

Mission of the Institute

1. Develop globally competent professionals for future talent requirements
2. Nurture a culture of research, innovation and entrepreneurship
3. Promote collaborations, extension and outreach programs for addressing industrial and societal needs
4. Imbibe moral and ethical values
5. Foster ecological and environmental consciousness

BVVS

Basaveshwar Engineering College, Bagalkote 587102
Department of Electronics and Computer Engineering

Vision of the Department

To be recognized as a Center of Excellence for Electronics and Computer Engineering education and research, developing competent professionals with moral and ethical values

Mission of the Department

1. Develop professionals with strong fundamentals, technical skills, committed to life-long learning
2. Create an environment for interdisciplinary research by collaborating with industries and institutes of higher learning
3. Imbibe high moral and ethical values

PSOs of the Department

1. Inculcate analytical, logical, programming, and managerial abilities
2. Apply software engineering principles and practices to provide software solutions
3. Design and develop computational systems for real-life applications

PEOs of the Department

1. Apply knowledge of electronics and computing to design and develop intelligent products or systems, including IoT, robotics, and AI-based applications, addressing societal and industrial needs.
2. Develop expertise in designing, analyzing, and implementing electronic circuits, embedded systems, and signal processing applications to meet the needs of real-world challenges.
3. Impart proficiency in programming languages, software development, and modern computational tools to create innovative solutions for complex engineering problems.
4. Adapt to emerging technologies in electronics and computer engineering by engaging in continuous learning and contributing to sustainable and ethical technological advancements.

BVVS

Basaveshwar Engineering College, Bagalkote 587102
Department of Electronics and Computer Engineering

Program Outcomes

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

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

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

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

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

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

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

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

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

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

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

Knowledge and Attitude Profile (WK)

WK1: A systematic, theory-based understanding of the natural sciences applicable to the discipline and awareness of relevant social sciences.

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

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

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

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

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

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

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

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

BVVS

Basaveshwar Engineering College, Bagalkote
587102 Department of Electronics and Computer Engineering

Table 1: Breakdown of Credits suggested by VTU /AICTE

Total Credits for BE=160 (as per VTU/AICTE)

Sl.	Category	VTU	AICTE	BEC Present	BEC Revised
1.	HSMC: HSS (English:1, Kannada:1, UHV,: 1, EV:1), HRM:3 (Offered by Dept) :7 AEC (IDT: 1, IPBL:1, SS:2, IKS: 1, MOOCS: 3) = 08	14+8=22	15	9+10=19	7+8=15
2.	BSC: Basic Science Courses (Physics, Chemistry, Mathematics and Biology for Engineers)	22	23	22	18
3.	ESC/ETC: Engineering Science Courses (Basic Elect/Electronics /Computer/Mechanics/Workshop/ Drawing etc.)	18	17	18	16
4.	PCC: Professional Core Courses	57	61	56	65
5.	PEC: Professional Elective Courses relevant to the branch with at least one course either fully or partially supported by industry	12	12	12	12
6.	OEC: Open Electives Courses/ Subjects from other technical/Arts/Commerce (3 MOOCS + 6)	9	12	09	09
7.	Mini (2) and Major projects (9)/Industrial Internships (10)	20	20	24	21
8.	Mandatory Course: PE, Yoga, NSS, Bridge course Maths 1 and 2 (lateral Entry)	00	00	00	00
	Total	**160**	**160**	**160**	**160**

BVVS

Basaveshwar Engineering College, Bagalkote
587102 Department of Electronics and Computer Engineering

Table-2: Semester wise Breakdown of Credits

Sem.	BSC	ESC ETC	HSMC	AEC	PCC	PEC	OEC	Proj.	Int.	Total
I	08	06	01	01 (IPBL)	04					20
II	08	10	01	01 (IDT)						20
III	03				17					20
IV	03		01		16					20
V			01	02(SS)	09	03	03	02		20
VI				01(IKS)	13	03	03			20
VII			03		06	06		09		24
VIII				03 (MOOCS)			03 (MOOCS)		10	16
Tot.	22	16	07	08	65	12	09	11	10	160

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT

Department of Electronics and Computer Engineering

SCHEME OF TEACHING

Semester: I Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	BSC	1BMAE101C	Differential Calculus and Linear Algebra	3(42)	2(28)	0	44	03	03	120	50	50	100	04
2.	BSC(IC)	1BCHE102C	Applied Chemistry for Emerging Electronics and Futuristic Devices	3(42)	0	2(26)	44	05	03	120	50	50	100	04
3.	ETC	1BAIA103C	Introduction to AI and Applications	3(42)	0	0	42	03	03	90	50	50	100	03
4.	ESC	1BXXY104N	Engineering Science Course I	3(42)	0	0	42	03	03	90	50	50	100	03
5.	PCC	1BCSX105D	Programming Language Course	3(42)	0	2(26)	44	05	03	120	50	50	100	04
6.	HSMC	1BHSB106C	Communication Skills	1(14)	0	0	13	02	01	30	50	50	100	01
7.	HSMC	1BHSC107M	Indian Constitution & Engineering Ethics	1(14)	0	0	14	02	--	30	100	--	100	PP
8.	AEC	1BUEA108P	Innovation and Design Thinking Lab (**Project-based learning)**	Two Contact hours/week for interaction between the faculty and students				03	03	30	50	50	100	01
			Total	**17**	**02**	**06**					**450**	**350**	**800**	**20**

Engineering Science Courses-I					Programming Language Courses (PLC)				
Code	**Title**	**L**	**T**	**P**	**Code**	**Title**	**L**	**T**	**P**
1BCVA104N	Building Sciences & Mechanics	3	0	0	1BCSA105D	Introduction to C Programming (For non IT programmes)	3	0	2
1BEEA104N	Introduction to Electrical Engineering	3	0	0	1BCSB105D	Python Programming(for CSE and allied programmes)	3	0	0
1BECA104N	Introduction to Electronics and Communication Engineering	3	0	0					
1BMEA104N	Introduction to Mechanical Engineering	3	0	0					
1BISA104N	Essentials of Information Technology	3	0	0					

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT

Department of Electronics and Computer Engineering

SCHEME OF TEACHING

Semester: II Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	BSC	1BMAE201C	Calculus, Laplace Transform, and Numerical Techniques	3(42)	2(28)	0	44	03	03	120	50	50	100	04
2.	BSC(IC)	1BPHE202C	Quantum Physics and Electronic Sensors	3(42)	0	2(26)	44	05	03	120	50	50	100	04
3.	ESC	1BMEE203C	Computer-Aided Engineering Drawing for EEE stream	2(28)	0	2(26)	28	05	03	90	50	50	100	03
4.	ESC	1BXXA204N	Engineering Science Course-II	3(42)	0	0	42	03	03	90	50	50	100	03
5.	PSC /ESC	1BXXA205C	Programme Specific Courses	3(42)	0	0	42	03	03	90	50	50	100	03
6.	AEC (NCMC)	1BHSA206M	Soft Skills	1(14)	0	0	13	02	01	30	100	--	100	PP
7.	HSMC	1BHSA207C/ 1BHAB207C	Samskrutika Kannada/ Balake Kannada	1(14)	0	0	13	02	01	30	50	50	100	01
8.	AEC	1BXXA208P	Interdisciplinary Project-Based Learning	Two Contact hours/week for interaction between the faculty and students				03	03	30	50	50	100	01
9.	PSC/ESC	1BXXY209L	Program-Specific Course Lab	0	0	2(26)	--	02	02	30	50	50	100	01
			Total	**16**	**02**	**06**					**500**	**400**	**900**	**20**

Engineering Science Courses-II					Programme Specific Courses(PSC)				
Code	**Title**	**L**	**T**	**P**	**Code**	**Title**	**L**	**T**	**P**
1BCV205N	Introduction to Building Sciences	3	0	0	1BCVA205C	Engineering Mechanics	3	0	0
1BEEA205N	Introduction to Electrical Engineering	3	0	0	1BMEA205C	Elements of Mechanical Engineering	3	0	0
1BECA205N	Introduction to Electronics & Communication Engineering	3	0	0	1BEEA205C	Basic Electrical Engineering	3	0	0
1BMEA205N	Introduction to Mechanical Engineering	3	0	0	1BECA205C	Fundamentals of Electronics & Communication Engineering	3	0	0
1BISA205N	Essentials of Information Technology	3	0	0	1BCSA205C	Programming in C	3	0	0
					1BBTA205C	Elements of Biotechnology and Biomimetics	3	0	0

Engineering Science Course Lab(PSCL)				
Code	**Title**	**L**	**T**	**P**
1BCVA209L	Mechanics and Materials Lab	0	0	2
1BMEA209L	Elements of Mechanical Engineering Lab	0	0	2
1BEEA209L	Basic Electrical Engineering Lab	0	0	2
1BECA209L	Fundamentals of Electronics & Communication Engineering Lab	0	0	2
1BCSA209L	C Programming Lab	0	0	2
1BBTA209L	Elements of Biotechnology Lab	0	0	2

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: III Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	BSC	1BMAE301C	Mathematics for Electrical Science Stream	3(42)	0	0	42	03	03	90	50	50	100	03
2.	IPCC	1BUEA302C	Data Structures with C	3(42)	0	2(26)	44	05	03	120	50	50	100	04
3.	PCC	1BUEA303C	Digital Electronics	3(42)	0	0	42	03	03	90	50	50	100	03
4.	PCC	1BUEA304C	Computer Architecture and Organization	4(56)	0	0	58	03	03	120	50	50	100	04
5.	PCC	1BUEA305C	Analog Electronics and Linear ICs	3(42)	0	0	42	03	03	90	50	50	100	03
6.	PCCL	1BUEA306L	Digital Electronics Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
7.	PCCL	1BUEA307L	Microcontroller Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
8.	PCCL	1BUEA308L	Analog Electronics and Linear ICs Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
9.	MC	1BHSA360M	Yoga	Two Contact hours/week for interaction between the faculty and students				02	--	30	100	--	100	0
		1BHSB360M	National Service Scheme (NSS)											
		1BHSC360M	Physical Education											
Total				**16**	**00**	**08**					**500**	**400**	**900**	**20**

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: IV Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	PCC	1BUEA401C	Discrete Mathematics and Graph Theory	3(42)	0	0	42	03	03	90	50	50	100	03
2.	PCC	1BUEA403C	Embedded System Design using ARM	3(42)	0	0	42	03	03	90	50	50	100	03
3.	PCC	1BUEA404C	Design and Analysis of Algorithms	3(42)	0	0	42	03	03	90	50	50	100	03
4.	PCC	1BUEA405C	Object Oriented Programming with Java	2(28)	2(28)	0	28	03	03	90	50	50	100	03
5.	PCCL	1BUEA406L	ARM Processor Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
6.	PCCL	1BUEA407L	Algorithms Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
7.	PCCL	1BUEA408L	Python Laboratory	0	0	4(52)	02	03	03	60	50	50	100	02
8.	BSC	1BBTA402C	Biology For Engineers	3(42)	0	0	42	03	03	90	50	50	100	03
9.	HSMC	1BHSA424C	Universal Human Values	1(14)	0	0	13	02	01	30	50	50	100	01
10.	MC	1BHSA460M	Yoga	Two Contact hours/week for interaction between the faculty and students				02	--	30	100	--	100	00
		1BHSB460M	National Service Scheme (NSS)											
		1BHSC460M	Physical Education											
Total				**15**	**02**	**08**					**550**	**450**	**1000**	**20**

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: V Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	IPCC	1BUEA501C	Data Science and Visualization	3(42)	0	2(26)	44	05	03	120	50	50	100	04
2.	PCC	1BUEA502C	Digital System Design using Verilog	4(56)	0	0	58	03	03	120	50	50	100	04
3.	PCCL	1BUEA503L	Verilog Lab	0	0	2(26)	--	02	02	30	50	50	100	01
4.	PEC	1BUEX504E	Professional Elective Course	3(42)	0	0	42	03	03	90	50	50	100	03
5.	OEC	1BXX5XXN	Open Elective Course	3(42)	0	0	42	03	03	90	50	50	100	03
6.	PROJ	1BUEA505P	Mini Project	0	0	4(52)	02	03	03	60	50	50	100	02
7.	AEC	1BHSA522C	Quantitative Aptitude and Professional Skills	2(28)	0	0	26	03	03	60	50	50	100	02
8.	HSMC	1BBTA523C	Environmental Studies	1(14)	0	0	13	02	01	30	50	50	100	01
9.	MC	1BHSA560M	Yoga	Two Contact hours/week for interaction between the faculty and students				02	--	30	100	--	100	00
		1BHSB560M	National Service Scheme (NSS)											
		1BHSC560M	Physical Education											
			Total	**16**	**00**	**08**					**500**	**400**	**900**	**20**

Professional Elective Course			
1BUEA504E	IoT and Applications	1BUEC504E	Operating Systems
1BUEB504E	Principles of Communication System	1BUED504E	Finite Automata and Formal Languages

Open Elective Course	
1BUEA531N	Electrical Sensors and Transducers

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: VI Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	IPCC	1BUEA601C	Machine Learning	3(42)	0	2(26)	44	05	03	120	50	50	100	04
2.	PCC	1BUEA602C	CMOS VLSI Design	4(56)	0	0	58	03	03	120	50	50	100	04
3.	IPCC	1BUEA603C	Data Base Management Systems	3(42)	0	2(26)	44	05	03	120	50	50	100	04
4.	PEC	1BUEX604E	Professional Elective Course	2(28)	0	2(26)	28	05	03	90	50	50	100	03
				3(42)	0	0	42	03	03					
5.	OEC	1BXX6XXN	Open Elective Course	3(42)	0	0	42	03	03	90	50	50	100	03
6.	PCCL	1BUEA605L	CMOS VLSI Laboratory	0	0	2(26)	--	02	02	30	50	50	100	01
7.	AEC	1BHSA600C	Indian Knowledge System	1(14)	0	0	13	02	01	30	50	50	100	01
8.	MC	1BHSA660M	Yoga	Two Contact hours/week for interaction between the faculty and students				02	--	30	100	--	100	00
		1BHSB660M	National Service Scheme (NSS)											
		1BHSC660M	Physical Education											
			Total	**19**	**00**	**02**					**450**	**350**	**800**	**20**

Professional Elective Course			
1BUEA604E	UI/UX Design	1BUEC604E	Unix Shell Programming
1BUEB604E	Full Stack Development	1BUED604E	Cyber Security

Open Elective Course	
1BUEA631N	Virtual Instrumentation

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: VII Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits © = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	HSMC	1BHSA711C	Research Methodology and IPR	3(42)	0	0	42	03	03	90	50	50	100	3
2.	IPCC	1BUEA702C	Big Data Analysis	3(42)	0	0	42	03	03	90	50	50	100	3
3.	IPCC	1BUEA703C	Computer Networks	2(28)	0	2(26)	28	05	03	90	50	50	100	3
4.	PEC	1BUEX704E	Professional Elective Course 1	3(42)	0	0	42	03	03	90	50	50	100	3
				2(28)	0	2(26)	28	05	03					
5.	PEC	1BUEX705E	Professional Elective Course 2	3(42)	0	0	42	03	03	90	50	50	100	3
				2(28)	0	2(26)	28	05	03					
				2(28)	2(28)	0	28	03	03					
6.	PROJ	1BUEA706P	Major Project	Two Contact hours/week for interaction between the faculty and students				03	03	270	50	50	100	9
			Total	**20**	**02**	**06**					**300**	**300**	**600**	**24**

Professional Elective Course 1			
1BUEA704E	System Verilog	1BUEC704E	Cloud Computing
1BUEB704E	Block Chain Technology	1BUED704E	Software Engineering and Project Management
Professional Elective Course 2			
1BUEA705E	Data Analytics using R	1BUEC705E	Cryptography and Network Security
1BUEB705E	Robotics and Automation	1BUED705E	Compiler Design

2025-26 Batch

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOT
Department of Electronics and Computer Engineering
SCHEME OF TEACHING

Semester: VIII Semester (B.E.)

Sl. No.	Course Category	Course Code	Course Title	Teaching-Learning Scheme				Examination Duration (Hrs/Sem.)		Total Hours	Examination Scheme and Evaluation			Credits (C) = Total Hours/30
				L Hrs/Week (Hrs/Sem.)	T Hrs/Week (Hrs/Sem.)	P Hrs/Week (Hrs/Sem.)	Term Work (TW) and Self Learning (SL) (TW+SL) Hrs/Sem.	CIE	SEE		CIE Marks	SEE Marks	Total Marks	
1.	AEC	1BUEXXXXO	Ability Enhancement Course (Online Courses)	3(42)	0	0	42	03	03	90	50	50	100	3
2.	OEC	1BUEXXXXO	Open Elective Course (Online Courses)	3(42)	0	0	42	03	03	90	50	50	100	3
3.	INT	1BUEA703I	Internship (Industry/Research) (12 - 14 weeks)	Two Contact hours/week for interaction between the faculty and students				03	03	300	50	50	100	10
Total											**150**	**150**	**300**	**16**

Third Semester Syllabus

Course Code		**:**	**1BMAE301C**	**Mathematics for Electrical Science Stream**	**Semester**	**:**	**03**
L:T:P		**:**	**3 : 0 : 0**		**Course Type**	**:**	**BSC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs**		**SEE Marks**	**:**	**50**
	Exam	**:**	**6Hrs**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs**		**Credits**	**:**	**03**

Professional Competency:

Applying statistical analysis, calculus, and transform methods to solve engineering problems.

Course Outcomes:

After the completion of the course student will be able to

CO1: Apply correlation and regression analysis to fit a suitable mathematical model for Statistical data

CO 2: Apply the concepts of probability distributions and Joint probability distribution for Engineering problems

CO 3: Apply Fourier series to study the behavior of periodic functions and their applications.

CO4: Apply Fourier transforms and Z-Transform to solve engineering problems.

UNIT-I	**10 Hrs.**

Curve fitting, Correlation and Regressions: Principles of least squares, Curve fitting by the method of least squares in the form $y = a + bx$, $y = a + bx + cx^2$, $y = ab^x$. Correlation, Regression and rank correlation.

UNIT–II	**11 Hrs.**

Probability Distributions: Random variables-discrete and continuous Probability distribution function, cumulative distribution function, Mathematical Expectation, mean and variance, Binomial Poisson's and Normal distribution (without proofs for mean and SD) – Problems. **Joint Probability Distribution:** Joint probability distributions for discrete and continuous random variable expectation, covariance and correlation.

UNIT–III	**11 Hrs.**

Fourier series and practical harmonic analysis: Periodic functions, Dirichlet's condition.Fourier series expansion of functions with period 2 and with arbitrary period $2l$. Half-range Fourier series. Practical harmonic analysis

UNIT–IV	**10 Hrs.**

Fourier and Z Transforms: Infinite Fourier transforms, Fourier cosine and sine transforms, Inverse Fourier transforms, Inverse Fourier cosine and sine transforms. Z-transforms.

Text Books

1. Ronald E. Walpole, Raymond H Myers, Sharon L Myers & Keying Ye "Probability & Statistics for Engineers & Scientists", Pearson Education, 9th edition, 2017.
2. Peter Bruce, Andrew Bruce & Peter Gedeck "Practical Statistics for Data Scientists" O' Reilly Media, Inc., 2nd edition 2020.

Reference Books

1. Erwin Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons,9th Edition , 2006.
2. B. S. Grewal "Higher Engineering Mathematics", Khanna publishers, 44th Ed., 2021.

3. G Haribaskaran " Probability, Queuing Theory & Reliability Engineering", Laxmi Publication , Latest Edition, 2006
4. Irwin Miller & Marylees Miller, John E. Freund's "Mathematical Statistics with Applications" Pearson. Dorling Kindersley Pvt. Ltd. India, 8th edition, 2014.
5. S C Gupta and V K Kapoor , "Fundamentals of Mathematical Statistics", S Chand and Company , Latest edition.
6. Robert V. Hogg, Joseph W. McKean & Allen T. Craig. "Introduction to Mathematical Statistics", Pearson Education 7thedition, 2013.
7. Jim Pitman. Probability, Springer-Verlag , 1993.
8. Sheldon M. Ross, "Introduction to Probability Models"11th edition. Elsevier, 2014.
9. A.M. Yaglomand I. M. Yaglom, "Probability and Information" D. Reidel Publishing Company. Distributed by Hindustan Publishing Corporation (India) Delhi,1983.
10. P.G. Hoel , S.C. Portand C. J. Stone, "Introduction to Probability Theory", Universal Book Stall, (Reprint), 2003.
11. S. Ross, "A First Course in Probability", Pearson Education India, 6th Ed.,2002.
12. W. Feller, "An Introduction to Probability Theory and its Applications", Vol. 1,Wiley, 3rdEd., 1968.
13. N. P. Bali and Manish Goyal , A Text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010

Web Links and Video Lectures:

https://drive.google.com/file/d/1KtCtZmnJnOvV1S68wXS9FZPXUFvRktBQ/view?usp=sharing

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	–	–	–	–	1 (WK 8)	–	–	1 (WK8)	3	2	1
CO2	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	1 (WK8)	–	–	–	1 (WK 8)	–	–	1 (WK8)	3	2	2
CO3	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	–	–	–	1 (WK 8)	–	–	2 (WK8)	3	3	3
CO4	3(WK2, WK4)	2 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	–	–	–	1 (WK 8)	–	–	2 (WK8)	3	3	2

Course Code		**:**	**1BUEA302C**	**Data Structures with C**	**Semester**	**:**	**03**
L:T:P		**:**	**3:0:2**		**Course Type**	**:**	**IPCC**
Hours/Sem.	**Teaching**	**:**	**42+26=68 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**44 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120 Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Develop efficient software solutions by selecting, implementing, and analyzing appropriate linear and nonlinear data structures to solve computational problems effectively.

Course Outcomes:

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

CO1: Explain the fundamental concepts of arrays, pointers, structures, unions and dynamic memory allocation, and demonstrate their usage in programming contexts.

CO2: Illustrate the organization and working principles of linear data structures (such as stacks and queues) and nonlinear data structures (such as trees).

CO3: Design and implement programs using linear and nonlinear data structures to solve computational problems effectively.

CO4: Analyze problem requirements to identify and justify the choice of appropriate data structures for optimized solutions.

Unit-I	**10 Hrs.**

Introduction to Data Structures: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations Review of pointers and dynamic Memory Allocation, ARRAYS and STRUCTURES: Arrays, Dynamic Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, representation of Multidimensional Arrays, Strings. **Stacks:** Stacks, Stacks Using Dynamic Arrays, Evaluation and conversion of Expressions.

Unit-II	**11 Hrs.**

Queues: Queues, Circular Queues, Using Dynamic Arrays, Multiple Stacks and queues. **Linked Lists:** Singly Linked, Lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials, Additional List Operations, Sparse Matrices, Doubly Linked List.

Unit-III	**10 Hrs.**

Trees: Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees. Binary Search trees, Selection Trees, Forests, Representation of Disjoint sets, Counting Binary Trees. **Priority Queues:** Single and double ended Priority Queues, Leftist Trees.

Unit-IV	**11 Hrs.**

Introduction to Efficient Binary Search Trees: Optimal Binary Search Trees. **Graphs:** The Graph Abstract Data Types, Elementary Graph Operations.

List of Experiments

1. Write a C program to implement iterative and recursive binary search algorithms. Define and use a macro to compare two integers in your program.
2. Write a C program to find the fast transpose of a sparse matrix.
3. Write a C program to implement a circular queue using dynamically allocated array and perform the following operations on it. (i) Insert an item (ii) Delete an item (iii) Display a circular queue.

4. Design, Develop and Implement a Program in C for the following Stack Applications for Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^ .
5. Write a C program Implementation of single ended priority queue.
6. Write a C program to implement a double ended priority queue.
7. Write a C program to implement a doubly linked circular list with a header node and perform the following operations on it. (i) Insert a node (iii) Display a doubly linked circular list in forward direction (ii) Delete a node (iv) Display a doubly linked circular list in reverse direction.
8. Write a C program to implement multiple linked queues (at least 5) and perform the following operations on them. (i) Add an item in ith queue (ii) Delete an item from ith queue (iii) Display ith queue
9. Write a C program to implement a binary search tree using linked representation and perform the following operations on it. (i) Insert an item (ii) Search an item (iii) In order Traversal
10. Write a C program to perform depth first search of a graph represented as an adjacency list.

Reference Books:

1. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Edition, McGraw Hill, 2014.
3. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning, 2014.
4. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
5. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Edition, McGraw Hill, 2013.
6. A. M. Tenenbaum, Data Structures using C, PHI, 1989.
7. Robert Kruse, Data Structures and Program Design in C, 2nd Edition, PHI, 1994.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	–	2 (WK2, WK6)	–	–	–	–	–	1 (WK4)	3	2	1
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	1 (WK8)	2 (WK2, WK6)	–	–	–	–	–	1 (WK4)	3	2	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	3	3
CO4	2 (WK2, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	2 (WK2, WK6)	–	–	–	1	–	2 (WK4, WK8)	3	3	2

Course Code		**:**	**1BUEA303C**	**Digital Electronics**	**Semester**	**:**	**03**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, analyze, and implement digital electronic circuits using Boolean logic, combinational and sequential circuit design principles, and modern simulation tools to solve engineering problems in computing and embedded systems.

Course Outcomes:

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

CO1: Simplify the Boolean functions and convert the expression in to required form.

CO2: Design a combinational logic circuit and understand the bistable memory elements.

CO3: Analyze and design synchronous and asynchrounous counters and shift registers.

CO4: Design synchronous and asynchrounous sequential circuits.

Unit-I	**10 Hrs.**

Introduction: Overview of Boolean algebra, De-Morgan's theorem, Canonical form expressions for digital functions, Review of basic logic gates, Positive and negative logic**. Generation of Switching Equations from Truth Tables**: Sum-of-products (SOP) method, Truth table to Karnaugh Map, Karnaugh simplifications (3 and 4 variable maps), Don't - care conditions, Product-of-sums (POS) method, Product-of-sums simplifications, Simplification by Quine-McClusky method.

Unit-II	**11 Hrs.**

Combinational Logic Circuits: Multiplexers, Demultiplexers, 1-of-16 Decoder, BCD to decimal decoders, Seven segment decoders, Encoders, Parity generators and checkers, Magnitude comparator, Programmable array logic, Programmable logic arrays. Arithmetic building blocks: Half-adder, Full adder, Parallel adder and subtractor, Look ahead carry adder, Arithmetic logic unit. **Sequential Circuits:** Flip-Flops: RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS flip-flop, Edge-triggered D flip-flop, T flip-flop, Edge-triggered JK flip-flop, JK Master-slave Flip-flops.

Unit-III	**11 Hrs.**

Sequential Circuits: Shift Registers: Serial in - Serial out, Serial in - Parallel out, Parallel in - Serial out, Parallel in - Parallel out, Universal shift register, Applications of shift registers. Counters: Asynchronous counters, Use of decoding gates for truncation, Synchronous counters for given modulus and sequence, Decade counters, Pre-settable counters, Counter design as a synthesis problem, A digital clock design.

Unit-IV	**10 Hrs.**

Design of synchronous and Asynchronous Sequential Circuits: Mealy and Moore models, State machine design, State transition diagram, State synthesis table, Design equations and circuit diagram, Algorithmic state machines and design examples.

Reference Books:

1. Donald P. Leach, Albert P. Malvino, Goutam Saha, "Digital Principles and Applications," 8th Edition, Tata McGraw Hill, 2015.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	3 (WK2, WK3, WK4, WK6)	–	-	–	–	–	–	–	1 (WK8)	3	2	1
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	1 (WK2, WK3, WK4, WK6)	-	–	–	–	–	–	1 (WK8)	3	2	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	-	–	–	1	1	–	2 (WK8)	3	3	3
CO4	3 (WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	-	–	–	–	1	–	2 (WK8)	3	3	2

Course Code		**:**	**1BUEA304C**	**Computer Architecture and Organization**	**Semester**	**:**	**03**
L:T:P		**:**	**4:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**56 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**58Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Design and analyze computer systems by applying concepts of processor architecture, memory organization, input/output systems, and performance evaluation to develop efficient and reliable computing solutions.

Course Outcomes:

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

CO1: Describe the fundamentals of machine instructions, addressing modes and Processor performance.

CO2: Explain the approaches involved in achieving communication between processor and I/O devices.

CO3: Analyze internal Organization of Memory and Impact of cache/Pipelining on Processor Performance.

CO4: Identify the working of Embedded systems and its processor Families.

Unit-I **14 Hrs.**

Basic Structure of Computers: Functional Units, Basic Operational Concepts, Bus Structure, Performance-Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. MachineInstructions and Programs: Memory Location and Addresses, Memory Operations, Instruction and Instruction sequencing, Addressing Modes.**Input/output Organization:** Accessing I/O Devices, Interrupts – Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices.

Unit-II **14 Hrs.**

Direct Memory Access: Bus Arbitration, Speed, Size and Cost of Memory Systems, Cache Memories-Mapping Functions. **Basic Processing Unit:** Some Fundamental Concepts: Register Transfers, Performing ALU operations, fetching a word from Memory, Storing a word in memory. Execution of a Complete Instruction

Unit-III **14 Hrs.**

Pipelining: Basic concepts, Role of Cache memory, Pipeline Performance.**Embedded Systems:** Examples of embedded Systems, Microwave Oven, Digital Camera, Home Telemetry, Processor Chips for Embedded Applications.

Unit-IV **14 Hrs.**

Embedded Processor Families: Microcontroller Based on the Intel 8051, Motorola Microcontrollers, ARM Microcontrollers. **Large Computer Systems:** Forms of Parallel Processing, Array Processors, The Structure of General-PurposeMultiprocessors.**Memory Organization in Multiprocessors, Multicomputers:** Local Area Networks, Ethernet(CSMA/CD) BUS, Token Ring, Network of Workstations

Reference Books:

1. Carl Hamcher, ZvonkoVranesic, SafwatZaky, “Computer Organization”, 5th Edition, Tata McGraw Hill.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	3 (WK2, WK3, WK4, WK6)	–	-	3 (WK8)	–	–	–	–	1 (WK8)	2	1	1
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	1 (WK2, WK3, WK4, WK6)	-	3 (WK8)	–	–	–	–	1 (WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	-	2 (WK8)	–	–	–	–	2 (WK8)	2	1	3
CO4	3 (WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	-	3 (WK8)	–	–	–	–	2 (WK8)	2	1	2

Course Code		**:**	**1BUEA305C**	**Analog Electronics and Linear ICs**	**Semester**	**:**	**03**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to analyze, design, and test analog electronic circuits using BJTs, FETs, Op-Amps, and ICs. Apply electronic concepts to solve basic engineering problems and evaluate circuit performance.

Course Outcomes:

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

CO1: Explain and analyze BJT biasing techniques, small-signal models, and power amplifier circuits.

CO2: Analyze FET biasing methods, small-signal amplifier configurations, and frequency response characteristics**.**

CO3: Describe the internal architecture, characteristics, and DC/AC amplifier applications of operational amplifiers.

CO4: Design, analyze, and test Op-Amp based applications, active filters, voltage regulators, and IC 555 timer circuits.

Unit-I	**10 Hrs.**

BJT Biasing: DC load line and bias point, Base bias, Collector-to-base bias, Voltage-divider bias, Bias circuit design, More bias circuits, Thermal stability of bias circuits, Biasing BJT switching circuits, **BJT Small Signal Analysis:** AC load lines, Transistor models and parameters, Common emitter circuit analysis, CE circuit with unbypassed emitter resister, common collector circuit analysis, common base circuit analysis, Comparison of CE, CC, CB circuits, **Power amplifiers:** Definition and amplifier types, series fed class-A amplifier, Transformer coupled class-A amplifier, class-B operation.

Unit-II	**10 Hrs.**

FET Biasing: Self bias and voltage divider bias arrangements of depletion type MOSFET, feedback biasing and voltage divider biasing arrangements of enhancement type MOSFET**, FET Small Signal Analysis:** Coupling**,** Bypassing and AC load lines, FET models and parameters, Common source circuit analysis, CS circuit with unbypassed source resistor, Common drain circuit analysis, Common gate circuit analysis, Frequency response of FET circuits.

Unit-III	**11 Hrs.**

Operational Amplifier Fundamentals: Op-Amp internal architecture block diagram, Overview of Op-Amp parameters: CMRR and PSRR, Offset voltage and offset current, Input and output impedance, Slew rate and frequency limitations, **Overview of Op-Amps as DC Amplifiers:** Non-inverting amplifiers, Inverting amplifiers, Difference amplifiers, **Op-Amps as AC Amplifiers:** Capacitor coupled voltage follower, High input impedance-capacitor coupled voltage follower. Capacitor coupled non inverting amplifiers, High input impedance-capacitor coupled non inverting amplifiers.

Unit-IV	**11 Hrs.**

Op-Amp Applications: Precision positive rectifiers, Sample and hold circuits. Differentiator circuit, Integrator circuit, Comparator and Zero crossing detector, Schmitt trigger circuit, First order and second order active Butterworth low-pass and high pass filters, **Voltage Regulators:** 78XX and 79XX fixed voltage regulators, **Timer IC-555 and Application:** Internal architecture and pin functions, Basic timer circuit, 555

timer used as astable and mono-stable multi-vibrator, Introduction to Voltage Controlled Oscillator and Phase Locked Loop.

Reference Books:

1. David A. Bell "Electronic Devices and Circuits", 5th Edition, Oxford University Press.
2. Boylestad and Nashelsky, "Electronic Devices and Circuit Theory," 10th Edition, 2009.
3. D. Roy Choudhury, ShailB.Jain,"Linear Integrated Circuits," 4thEdition, Reprint 2006, NewAge International ISBN 978-81-224-3098-l.
4. Ramakant A. Gayakwad, "Op-Amps and Linear Integrated Circuits," 4thEdition, Pearson, 2015, ISBN81-7808-501-1.
5. Somanathan Nair, "Linear Integrated Circuits: Analysis, Design & Applications," 1stEdition, 2015, Wiley India.
6. James Cox, "Linear Electronic Circuits and Devices," Cengage Learning, Indian Edition, 2008, ISBN-13: 978-07-668-3018-7.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK2, WK3, WK4, WK6)	–	-	3 (WK8)	–	–	–	–	1 (WK8)	2	1	1
CO2	3 (WK1,WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	1 (WK2, WK3, WK4, WK6)	–	-	3 (WK8)	–	–	–	–	1 (WK8)	2	1	2
CO3	3 (WK1,WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	–	-	2 (WK8)	–	–	–	–	2 (WK8)	2	1	3
CO4	3 (WK1,WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	1 (WK2, WK6)	3 (WK8)	–	–	–	–	2 (WK8)	2	1	2

Course Code		**:**	**1BUEA306L**	**Digital Electronics Laboratory**	**Semester**	**:**	**03**
L:T:P*		**:**	**0:0:2**		**Course Type**	**:**	**PCCL**
Hours/Sem.	**Teaching**	**:**	**26 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**--**		**SEE Marks**	**:**	**50**
	Exam	**:**	**04 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**30 Hrs.**		**Credits**	**:**	**01**

Professional Competency:

Design, implement, simulate, and verify combinational and sequential digital circuits using standard digital ICs and modern design tools while adhering to engineering practices and laboratory safety.

Course Outcomes:

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

CO1: Verify the operation of basic logic gates and implement Boolean functions using standard digital ICs.

CO2: Design, implement, and test combinational logic circuits such as adders, subtractors, multiplexers, demultiplexers, encoders, and decoders

CO3: Design, implement, and analyze sequential logic circuits using flip-flops, counters, shift registers, and registers for given requirements

CO4: Demonstrate laboratory skills by conducting experiments, interpreting results, documenting observations, and working effectively in teams

List of Experiments

1. K-Map simplification and verification using digital gates.
2. Design and implementation of digital multiplexer and de-multiplexer.
3. Design and implementation of 3 bit binary to Gray and binary to Excess-3 converter.
4. Design and implementation of 1 and 2-bit comparators.
5. Design and verification of 3 bit encoders and decoders using gates.
6. Design of combinational circuit for BCD to decimal conversion to drive 7-segment display.
7. Truth table verification of flip flops D and JK flip flops (IC7474 and IC7476).
8. Shift registers using flip flop ICs.
9. Design and verification of 4 bit binary ring and Johnson counter using JK flip flop ICs.
10. Design and verification of 3 bit binary asynchronous counter using JK flip flop ICs.
11. Design and verification of 3 bit binary synchronous counter using JK flip flop ICs.
12. Design and verification of 3 bit synchronous counter for a given sequence using JK flip flop ICs.
13. Design of sequential circuit for a given state table/state diagram (Mealy/Moore model) using JK flip flops.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	3	1	2
CO2	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	2	1	2
CO3	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	3	2	3
CO4	3 (WK1, WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK6)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	3	2	3

Course Code		**:**	**1BUEA307L**	**Microcontroller Laboratory**	**Semester**	**:**	**03**
L:T:P		**:**	**0:0:2**		**Course Type**	**:**	**PCCL**
Hours/Sem.	**Teaching**	**:**	**26 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**--**		**SEE Marks**	**:**	**50**
	Exam	**:**	**04 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**30 Hrs.**		**Credits**	**:**	**01**

Professional Competency:

Design, implement, test, and validate 8051 microcontroller-based embedded systems by integrating hardware and software using modern development tools and standard engineering practices.

Course Outcomes:

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

CO1: Exhibit the Programming skills using Assembly Language and C**.**

CO2: Write Assembly Language Programs using 8051 for solving simple problems those manipulate input data using different instructions**.**

CO3: Interface various input and output devices to 8051 and control them through Programming**.**

CO4: Debug the programs.

List of Experiments

1. Data Transfer: Block Move, Exchange, Sorting, Finding largest element in an array.
2. Arithmetic Instructions – Addition/Subtraction, Multiplication and Division, Square, Cube.
3. BCD, Hex, Decimal Counters.
4. Boolean & Logical Instructions (Bit manipulations).
5. Conditional CALL & RETURN.
6. Code Conversion: BCD-ASCII; ASCII- Decimal-ASCII; HEX-Decimal and Decimal-HEX.
7. Programs to generate time delay.
8. Program to blink LED.
9. Program to work on relay and buzzer.
10. Program to display numbers from 000 to FFF on 7-segment Display.
11. Program to interface stepper, DC motors and rotate.
12. Program to interface LCD and display the text.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	3	1	2
CO2	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	2	1	2
CO3	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	3	2	3
CO4	3 (WK1, WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK6)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	3	2	3

Course Code		:	**1BUEA308L**	**Analog Electronics and Linear ICs Laboratory**	**Semester**	:	**03**
L:T:P		:	**0:0:2**		**Course Type**	:	**PCCL**
Hours/Sem.	**Teaching**	:	**26 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**--**		**SEE Marks**	:	**50**
	Exam	:	**04 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**30 Hrs.**		**Credits**	:	**01**

Professional Competency:

Develop the skills to design, build, and test basic analog electronic circuits using electronic components and ICs. Use laboratory instruments to measure circuit performance and analyze the results.

Course Outcomes:

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

CO1: Analyze and characterize rectifier circuits and Zener diode voltage regulators for basic power supply applications.

CO2: Design and implement BJT and Op-Amp based amplifier circuits and evaluate their performance parameters.

CO3: Design and realize Op-Amp based applications including precision rectifiers, comparators, Schmitt triggers, and active filters.

CO4: Implement and analyze fixed-voltage regulator circuits and IC 555 timer based astable and monostable multivibrators.

List of Experiments

1. Full wave bridge rectifier with filter and measure the ripple factor.
2. Characteristics of Zener diode and design a simple Zener voltage regulator; determine line and load regulation.
3. Design and set up the BJT common emitter voltage amplifier with and without feedback and determine the gain-bandwidth product, input impedance and output impedance.
4. To study the operation of a common source FET amplifier and determine voltage gain, input impedance and output impedance.
5. Implementation of Inverting, non-inverting and differential DC amplifier for a given gain using Op-Amp IC.
6. Implementation of capacitor coupled non-inverting AC amplifier using Op-Amp IC.
7. Implementation of full-wave precision rectifier using Op-Amp IC.
8. Implementation of Zero-Crossing Detector (ZCD) and Schmitt Trigger using Op-Amp IC.
9. Implementation of Second-Order Butterworth Low-Pass and High-Pass Filters using Op-Amp IC.
10. Realization of +5V, +12V, -5V, -12V regulated power supply using IC78XX and IC79XX.
11. Implementation of Astable and Monostable Multivibrators using IC 555 Timer.
12. Realization of VCO using IC-565.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK2, WK3, WK4, WK6)	1 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	3	2	-
CO2	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK8)	3	3	1
CO3	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	3	3	2
CO4	2 (WK1, WK2, WK4)	2 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK6)	3 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	3	2	2

Fourth Semester Syllabus

Course Code		**:**	**1BUEA401C**	**Discrete Mathematics and Graph Theory**	**Semester**	**:**	**04**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Analyze, model, and solve engineering and computational problems using discrete mathematical structures, graph theory, and algorithmic techniques.

Course Outcomes:

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

- **CO1:** Understand and apply the logic of mathematics in the field of computer science.
- **CO2:** Explain and analyse different relations and their closures, Posets and lattices.
- **CO3:** Apply theory of solution of recurrence relations to solve them.
- **CO4:** Apply the concepts related to graph theory and different algorithms for solving the problems related to trees.

Unit-I **11 Hrs.**

Fundamentals of Logic: Basic connectives and Truth tables, Logical equivalence-Laws of Logic, Logical Implication-Rules of Inference. Quantifiers- Universal and Existential Quantifiers. **Relations:** Types and Properties of Relations (revision), Binary Relations and Their Applications.

Unit-II **11 Hrs.**

Relations: Computer Recognition-Zero One Matrices and Directed graphs, Transitive closure, Equivalence relation and Partitions, Posets and Hasse Diagrams, Lattices **Recurrence Relations:** Definition, Homogeneous recurrence relations. Non-homogeneous recurrence relations.

Unit-III **10 Hrs.**

Solutions to Recurrence Relations: Solution of homogeneous and non-homogeneous recurrence relations. Generating functions. Solution of recurrence relation by generating function. **Introduction to Graph Theory:** Definitions and Examples, Sub-graphs, Complements, and Graph Isomorphis. Vertex Degree: Euler Trails and Circuits, Planar Graphs, Hamilton Paths and Cycles.

Unit-IV **10 Hrs.**

Graph Theory-Trees: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes. **Optimization and Matching:** Dijkstra's Shortest Path Algorithm, Minimal Spanning Trees-The algorithms of Kruskal and Prim's.

Reference Books:

1. Kolman, Busby,Ross “Discrete Mathematical Structures,” 6thEdition, Prentice Hall of India, 2010 onwards.
2. Ralph P.Grimaldi, “Discrete and Combinational Mathematics,” 5thEdition, Pearson Education, 2011.
3. Kenneth Rosen, “Discrete Mathematics and Its Applications with Combinatorics and Graph

Theory," (SIE) 7th Edition onwards.
4. D. S. Chandrasekharaia, "Discrete Mathematical Structures, Prism, 2005.
5. D. S. Chandrasekharaiah, "Graph Theory and Combinatorics," Prism, 2005.

Web Links and Video Lectures (e-Resources):

1. https://archive.nptel.ac.in/courses/111/106/111106086/
2. https://onlinecourses.nptel.ac.in/noc22_ma10/preview

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	2 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	1	1	2
CO2	2 (WK1, WK2, WK3, WK4)	2 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	1	1	2
CO3	2 (WK1, WK2, WK3, WK4)	2 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	1	1	2
CO4	2 (WK1, WK2, WK4)	2 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	1 (WK8)	–	–	1	1	2

Course Code		**:**	**1BUEA403C**	**Embedded System Design using ARM**	**Semester**	**:**	**04**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, implement, interface, and optimize ARM-based embedded systems using modern hardware and software tools to solve real-world engineering problems

Course Outcomes:

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

CO1: Comprehend the ARM Architectural features and Instructions.

CO2: Develop programs using ARM instruction set for an ARM Microcontroller.

CO3: Analyze C-Compiler Optimizations and portability issues in ARM Microcontroller.

CO4: Apply the concepts of Exceptions and Interrupt handling mechanisms in developing applications.

Unit-I	**11 Hrs.**

ARM Embedded Systems: The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software. **ARM Processor Fundamentals:** Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table, Core Extensions.

Unit-II	**11 Hrs.**

Introduction to the ARM Instruction Set: Data Processing Instructions, Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Co-processor Instructions, Loading Constants.

Unit-III	**10 Hrs.**

C Compilers and Optimization: Basic C Data Types, C Looping Structures, Register Allocation, Function Calls, Pointer Aliasing, Portability Issues.

Unit-IV	**10 Hrs.**

Exception and Interrupt Handling: Exception handling, ARM processor exceptions and modes, Vector table, Exception priorities, Link register offsets, Interrupts, Assigning interrupts, Interrupt latency, IRQ and FIQ exceptions, Basic interrupt stack design and implementation**.**

Reference Books:

1. Andrew N. Sloss, Dominic Symes and Chris Wright, “ARM System Developers Guide”, Elsevier, 2008, Morgan Kaufman publishers.
2. Raghunandan G. H., “Microcontroller (ARM) and Embedded System”, 2019, Cengage learning Publication.
3. Insider’s Guide to the ARM7 based Microcontrollers, Hitex Ltd.,1st Edition, 2005.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	2 (WK1, WK2, WK3, WK4)	–	–	–	2 (WK2, WK6)	–	–	–	–	–	3 (WK8)	3	1	2
CO2	2 (WK1,WK2, WK3, WK4)	–	3 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	1	2
CO3	2 (WK1,WK2, WK3, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	1	1
CO4	2 (WK1,WK2, WK4)	3 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	–	2 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	1	1

Course Code		**:**	**1BUEA404C**	**Design and Analysis of Algorithms**	**Semester**	**:**	**04**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, analyze, and implement efficient algorithms using appropriate algorithmic paradigms and data structures to solve complex computational problems while optimizing time and space complexity.

Course Outcomes:

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

CO1: Apply asymptotic notational methods to analyze the performance of the algorithms in terms of time complexity.

CO2: Demonstrate divide & conquer approaches and decrease & conquer approaches to computational problems.

CO3: Make use of transform & conquer and dynamic programming design approaches to solve the given real world or complex computational problems**.**

CO4: Apply greedy and input enhancement methods to solve graph & string based computational problems.

Unit-I	**10 Hrs.**

Introduction: What is an Algorithm?, Fundamentals of Algorithm Problem Solving. Fundamentals of the Analysis of Algorithm Efficiency: Analysis Framework, A symptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms.

Unit-II	**11 Hrs.**

Decrease-And Conquer: Insertion Sort, Topological Sorting. **Divide And Conquer:** Merge Sort, Quick Sort, Binary Tree Traversals, Multiplication of Large Integers and Strassen's Matrix Multiplication.

Unit-III	**10 Hrs.**

Space-TimeTradeoffs: Sorting by Counting: Comparison counting sort, Input Enhancement in String Matching: Horspool's Algorithm. Dynamic Programming: Three basic examples, The Knapsack Problem and Memory Functions, Warshall's and Floyd's Algorithms.

Unit-IV	**11 Hrs.**

The Greedy Method: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm ,Huffman Trees and Codes. **Limitations Of Algorithmic Power:** Decision Trees, P, N P, and NP- Complete Problems. CO PING With Limitations Of Algorithmic Power: Back tracking (n-Queens problem, Subset-sum problem), Branch and Bound (Knapsack problem), Approximation algorithms for NP-Hard problems (Knapsack problem).

Reference Books:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition(Indian), 2017.
2. Ellis Horowitz, Satraj Sahni, Rajasekaran, "Computer Algorithms/ C++", 2nd Edition, 2014, Universities Press.

3. Thomas H.Cormen, Charles E.Leiserson, Ronal L. Rivest, Clifford Stein, “Introduction to Algorithms”, 3rd Edition, PHI.
4. S. Sridhar, “Design and Analysis of Algorithms”, Oxford (HigherEducation).

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	1 (WK5)	2 (WK8)	2 (WK2, WK6)	–	–	–	–	–	2 (WK4, WK8)	3	2	1
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	1	–	2 (WK4, WK6)	3	3	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		:	**1BUEA405C**	**Object Oriented Programming with Java**	**Semester**	:	**04**
L:T:P		:	**2:2:0**		**Course Type**	:	**PCC**
Hours/Sem.	**Teaching**	:	**28+28=56 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**28 Hrs.**		**SEE Marks**	:	**50**
	Exam	:	**06 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**90 Hrs.**		**Credits**	:	**03**

Professional Competency:

Develop robust, modular, and reusable software applications using Java and object-oriented programming principles

Course Outcomes:

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

- **CO1:** Apply fundamental Java programming concepts including data types, operators, control structures, arrays, methods and basic program development.
- **CO2:** Implement core Object-Oriented Programming principles including classes, object construction, packages, interfaces, and lambda expressions.
- **CO3:** Write robust applications utilizing inheritance mechanisms, super classes, enumerations, and comprehensive exception handling techniques.
- **CO4:** Apply generic programming and collections framework concepts to develop Java applications.

Unit-I **10 Hrs.**

Fundamental Programming Structures: Dissecting the "Hello, World" Program; Compiling and Running a Java Program; Primitive Types; Variables; Arithmetic Operations; Strings; Input and Output; Control Flow; Arrays; Functional Decomposition.

Unit-II **10 Hrs.**

Object-Oriented Programming: Working with Objects; Implementing Classes; Object Construction; Static Variables and Methods, Packages; Nested Classes; Documentation Comments; Interfaces; Static, Default and Private Methods in interface; Lambda Expressions; Method and Constructor References; Local and Anonymous Classes.

Unit-III **11 Hrs.**

Inheritance and Exceptions: Extending a Class; Object: The Cosmic Superclass; Enumerations; Runtime Type Information and Resources; Exception Handling: Throwing Exceptions; The Exception Hierarchy; Declaring Checked Exceptions; Catching Exceptions; the Try-with-Resources Statement; The finally Clause; Re-throwing and Chaining Exceptions; Uncaught Exceptions and the Stack Trace.

Unit-IV **11 Hrs.**

Generic Programming and Collections: Generic Classes; Generic Methods; Type Bounds; Type Variance and Wildcards; Restrictions on Generics; an Overview of the Collections Framework; Iterators; Sets; Maps.

List of Experiments

1. The sieve of Eratosthenes is one of the most efficient ways to find all primes smaller than n when n is smaller than 10 million. Given a number n, use JAVA to print all primes.
2. The Gauss-Jordan method is also known as Gauss-Jordan elimination method is very useful in solving a linear system of equations. It is a technique in which a system of linear equations is resolved by the means of matrices. Develop a JAVA application.

3. To compute a square root of any positive number a, start with an initial guess x=x1 for √a; then calculate successive approximations x2,x3...,√a using the formula:

$$x_i = \frac{x_{i-1} + (a/x_{i-1})}{2}, i = 2, 3, \ldots$$

   Develop a JAVA application that implements the above SQRT function to compute the square root of any positive number.
4. Model a lamp as a Java object. Make a Lamp class. This will contain atleast one instance variable which will be of type Boolean and will hold the state of the lamp: i.e., whether it is on or off. In addition, add methods to do the following things: switch the light on and off, and check its current state, i.e., whether it is on or off. Maintain proper encapsulation mechanism.
   Next, write a launcher class with a main() method to carry out the following tasks:

   - create a lamp object;
   - turn it on and off;
   - Print the lamp's on/off status to the console.

5. Given the following functional interface:
   interface MathOperation {int operation(int a, int b);}

   Develop an application that would implement the above interface using lambda expressions as to perform theaddition, subtraction, multiplication and division operations.
6. The String class in JAVA has a static method compare To Ignore Case, which compares two strings and the Arrays class has a static sort method. Build a JAVA program that creates an array of strings, use the sort function from Arrays class to sort the strings by passing the compare To Ignore Case function as a parameter to the sort function using method reference. Print the sorted array.
7. XYZ technologies is firm that has 5 employees with 1 manager, and 4 technicians. XYZ wants to digitize its payroll system, the following requirements: Dearness Allowance is 70% of basic for all employees. House Rent Allowance is 30% of basic for all employees. Income Tax is 40% of gross salary for all employees. The annual increments to the employees are to be given of the following criteria: - Manager 10% of the basic salary, and Technicians 15% of basic. Develop the pay roll for XYZ. Implement a class hierarchy using inheritance, where Employee is an abstract class and Manager and Technician are derived from Employee. Demonstrate a polymorphic behavior for giving the annual increments.
8. Define a new Exception class named Odd Exception. Create a new class named Even Odd. Write a method called halfOf(), which takes an int as parameter and throws an Odd Exception if the int is odd or zero, otherwise returns (int / 2). Write a main method that calls halfOf() three times (once each with an even int, an odd int, and zero), with three try/catch blocks, and prints either the output of halfOf() or the caught Odd Exception.
9. Implement a class named Fraction that represents fractions with numerator and denominator always stored reduced to lowest terms. If fraction is negative, the numerator will always be negative, and all operations leave results stored in lowest terms. Implement the addition, subtraction, multiplication and division operation for the Fraction class
10. Create a class Student that has instance variables as Name, Age, Address and access transmutation methods to access the instance variables along with display method to print the details of student. Next write a main() function that will create a collection of 10 students and reverse the list. Print the details before and after reversing the collection.
11. Use generics to build a class Sort. Implement the bubble sort algorithm to sort an array of any type.

12. Write a generic method to count the number of elements in a collection that have a specific property (for example, odd integers, prime numbers, palindromes).

Reference Books:

1. Cay S. Horstmann, ”Core Java SE 9 for the Impatient”, 2nd Edition, 2018, Addison Wesley
2. Herbert Schild, “Java: The Complete Reference”, McGraw-Hill, 10th Edition, 2018.
3. David Gallardo, Ed Burnette, Robert Mcgovern, “Eclipse in Action a Guide for Java Developers”, Manning Publications, 2003.
4. Ed. Burnette, “Eclipse IDE Pocket Guide: Using the Full-Featured IDE”, O'Reilly Media, Inc, USA, 2005.
5. Cay S. Horstmann, ”Core Java Volume I—Fundamentals”, 10th Edition, 2015, Prentice Hall
6. Joshua Bloch, ”Effective Java”, 3rd Edition, 2017, Addison-Wesley Professional
7. Ken Kousen, “Modern Java Recipes”, 2017, O'Reilly Media, Inc.
8. Oracle Java Tutorials and Documentation, Oracle Corporation. (https://docs.oracle.com/javase/tutorial/)

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	2 (WK8)	–	–	2 (WK4, WK8)	3	2	1
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	2 (WK8)	1 (WK8)	–	2 (WK4, WK6)	3	3	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		:	**1BUEA406L**	**ARM Processor Laboratory**	**Semester**	:	**04**
L:T:P		:	**0:0:2**		**Course Type**	:	**PCCL**
Hours/Sem.	**Teaching**	:	**26 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**--**		**SEE Marks**	:	**50**
	Exam	:	**04 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**30 Hrs.**		**Credits**	:	**01**

Professional Competency:

Design, develop, interface, test, and troubleshoot ARM processor-based embedded systems using modern hardware, software, and debugging tools to solve real-world engineering problems

Course Outcomes:

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

CO1: Demonstrate knowledge of ARM processor features, instruction sets, and addressing modes.

CO2: Implement programs to interface GPIOs, timers, ADC/DAC.

CO3: Design, implement, and test programs to solve specific problem statements.

CO4: Collaborate in lab activities, prepare structured reports, and present outcomes with clarity and professionalism.

List of Experiments

1. Write a C program to interface the LEDs.
2. Write a C program to implement counter by interfacing 7 Segment display.
3. Write a C program to interface buzzer.
4. Write a C program to interface relay.
5. Write a C program to interface keys and keyboard.
6. Write a C program to interface stepper motor.
7. Write a C program to interface DC motor.
8. Write a C program to read the digital data from internal ADC.
9. Write a C program to generate various waveforms using internal DAC.
10. Write a C program to interface external interrupt source.
11. Write a C program to interface the LCD and display the given text in given fashion.
12. Write a C program to transmit the data serially using UART.

Reference Books:

1. Andrew N. Sloss, Dominic Symes and Chris Wright, “ARM System Developers Guide”, Elsevier, 2008, Morgan Kaufman Publishers.
2. Raghunandan G. H., “Microcontroller (ARM) and Embedded System”, 2019, Cengage learning Publication.
3. Insider’s Guide to the ARM7 based Microcontrollers, 1stEdition, 2005, Hitex Ltd.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	–	2 (WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK4, WK8)	1	1	1
CO2	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	1 (WK8)	–	1 (WK4, WK6)	1	1	1
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	2	2 (WK8)	–	1 (WK4, WK6, WK8)	1	1	1
CO4	–	–	2 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	3	3 (WK8)	–	2 (WK4, WK6, WK8)	1	1	1

Course Code		:	**1BUEA407L**	**Algorithms Laboratory**	**Semester**	:	**04**
L:T:P		:	**0:0:2**		**Course Type**	:	**PCCL**
Hours/Sem.	**Teaching**	:	**26 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**--**		**SEE Marks**	:	**50**
	Exam	:	**04 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**30 Hrs.**		**Credits**	:	**01**

Professional Competency:

Design, implement, analyze, and evaluate efficient algorithms using appropriate algorithmic paradigms and programming tools to solve computational problems while optimizing time and space complexity.

Course Outcomes:

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

CO1: Develop programs to solve computational problems using suitable algorithm design strategy.

CO2: Compare algorithm design strategies by developing equivalent programs and observing Running times for analysis (Empirical).

CO3: Make use of suitable integrated development tools to develop programs

CO4: Choose appropriate algorithm design techniques to develop solutions to the computational and complex problems.

List of Experiments

1. Design and implement Program to sort a given set of n integer elements using the Selection Sort method and compute its time complexity. Run the program for varied values of n>5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
2. Design and implement Program to sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n>5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
3. Design and implement Program to sort a given set of ninteger elements using Merge Sort method and compute its time complexity. Run the program for varied values of n>5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
4. Design and implementProgram to find Minimum Cost Spanning Tree of a givenconnected undirected graph using Kruskal'salgorithm.
5. Design and implementProgram to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim'salgorithm.
6. a. Design and implementProgram to solve All-Pairs Shortest Paths problem usingFloyd'salgorithm.
   b. Design and implementProgram to find the transitive closure using Warshall's algorithm.
7. Design and implementProgram to find shortest paths from a given vertex in a weighted connected graph to other vertices using Dijkstra's algorithm.
8. Design and implement Program to obtain the Topological ordering of vertices in a given digraph.

9. Design and implementProgram to solve 0/1 Knapsack problem using Dynamic Programming method.
10. Design and implement Program to solve discrete Knapsack and continuous Knapsack Problems using greedy approximation method.
11. Design and implementProgram to find a subset of a given set S={sl,s2,.....,sn} of nPositive integers whose sum is equal to a given positive integer.
12. Design and implement Program for NQueen's problem using Backtracking.

Reference Books:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", 3rd Edition (Indian),2017,Pearson.
2. Ellis Horowitz, Satraj Sahni and Rajasekaran, "ComputerAlgorithms/C++", 2nd Edition, 2014, Universities Press.
3. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, "Introduction to Algorithms", 3rdEdition, PHI.
4. S.Sridhar, "Design and Analysis of Algorithms," Oxford(HigherEducation)

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	3	3
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6, WK8)	3	3	2
CO3	2 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	1 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6)	3	3	2
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		**:**	**1BUEA408L**	**Python Laboratory**	**Semester**	**:**	**04**
L:T:P		**:**	**0:0:4**		**Course Type**	**:**	**PCCL**
Hours/Sem.	**Teaching**	**:**	**52 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**02**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**60 Hrs.**		**Credits**	**:**	**02**

Professional Competency:

Apply Python concepts such as data types, control structures, functions, modules, and object-oriented programming to develop efficient applications.

Course Outcomes:

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

CO1: Develop Python programs using fundamental programming constructs.

CO2: Apply functions and data structures to solve computational problems.

CO3: Design applications involving files, modules, and exception handling.

CO4: Develop object-oriented Python applications and perform basic data analysis and visualization.

List of Experiments

1. Python Fundamentals and Control Structures

- Introduction to Python and Python Environment, Variables, Data Types, Input and Output, Operators and Expressions, Conditional Statements, Iterative Statements (for, while), Pattern Generation and Basic Problem Solving

Laboratory

1. Installation and execution of Python programs.
2. Programs on arithmetic operations and expressions.
3. Programs using conditional statements.
4. Programs using loops.
5. Pattern generation programs.
6. Number-based applications (factorial, prime number, palindrome, Fibonacci series).

2. Functions, Strings and Data Structures

- Functions and Parameter Passing, Recursive Functions, Strings and String Operations, Lists and List Methods, Tuples, Sets and Dictionaries

Laboratory

1. Programs using user-defined functions.
2. Recursive function applications.
3. String manipulation programs.
4. List processing applications.
5. Dictionary-based applications.
6. Programs involving tuples and sets.

3. Files, Modules and Exception Handling

- File Handling Operations, Text and Binary Files, Modules and Packages, Exception Handling,

Regular Expressions

Laboratory

1. Reading and writing text files.
2. File processing applications.
3. Creating and using modules.
4. Programs using exception handling.
5. Applications involving regular expressions.
6. Mini applications using files and modules.

4. Object Oriented Programming and Application

- Classes and Objects, Constructors and Methods, Inheritance and Polymorphism, Introduction to NumPy and Matplotlib , Application Development Concepts

Laboratory

1. Programs using classes and objects.
2. Inheritance and polymorphism examples.
3. Numerical computations using NumPy.
4. Data visualization using Matplotlib.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	1 (WK5)	2 (WK8)	2 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6)	3	3	3
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2 (WK8)	2 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK6, WK8)	3	3	2
CO3	2 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	3 (WK8)	2 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK6)	3	3	2
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	2 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6, WK8)	3	3	3

Professional Competency:

	Course Code	:	**1BBTA402C**	**Biology for Engineers**	**Semester**	:	**04**
	L:T:P*	:	**3:0:0**		**Course Type**	:	**BSC**
Hours/Sem.	**Teaching**	:	**42 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**42 Hrs.**		**SEE Marks**	:	**50**
	Exam	:	**6 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**90 Hrs.**		**Credits**	:	**03**

Understanding fundamentals of biological processes and apply to solve engineering problems, bridging the technical engineering with living systems.

Course Outcomes:

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

CO1: Explain the structure and functions of cells and biomolecules, properties and applications of special biomolecules in engineering and industrial contexts.

CO2: Interpret bio-inspired models and natural mechanisms and correlate them with engineering designs and technological innovations.

CO3: Analyze the functional architecture of human organ systems as models to design and evaluate engineering solutions**.**

CO4: Apply the concept of bioprinting, artificial intelligence, and microbial processes in the design and development of innovative solutions for healthcare, food science, andsustainable infrastructure.

UNIT-I	**10 Hrs.**

Introduction to Biology: The cell: Structure, and functions of a cell. Biomolecules: Properties and functions of Carbohydrates, Nucleic acids, Proteins and Lipids. Importance of special biomolecules: Enzymes, vitamins and hormones -properties and functions. **Biomolecules and their Applications:** Carbohydrates in bioplastics production, Proteins in food production, Lipids in biodiesel.

UNIT-II	**10 Hrs.**

Bio Inspiration Models Used in Engineering: Bio inspiration - Introduction, Alliance between Engineering and Biology, Biomimicry - Science mimicking nature. Bird flying (GPS and aircrafts), Lotus leaf effect (Super hydrophobic and self-cleaning surfaces), Gecko Feet, Plant burrs (Velcro), Shark skin (Friction reducing swimsuits), Kingfisher beak (Bullet train), Fire fly LED. **Nature Bio inspired Mechanisms:** BioEcholocation (ultrasonography, sonars), Photosynthesis (photovoltaic cells, bionic leaf), Respiration (MFCs) .

UNIT-III	**12 Hrs.**

Human Organ Systems and Bio Designs: Brain as a CPU system -architecture, CNS and Peripheral Nervous System, signal transmission, EEG. Engineering solutions for Parkinson's disease.Heart as a pump system -architecture, electrical signalling - ECG monitoring and heart related issues, reasons for blockages of blood vessels, design of stents, pace makers, defibrillators. Lungs as purification system gas exchange mechanisms, spirometry, Ventilators.Eye as a Camera system, bionic eye. Kidney as a filtration system - dialysis systems.

UNIT-IV	**10 Hrs.**

Trends In Bioengineering: Bioprinting techniques and materials, 3D printing of ear, bone and skin, electrical tongue and electrical nose in food science, Self-healing bioconcrete (based on bacillus spores, calcium lactate nutrients and biomineralization processes), Biomining via microbial surface adsorption. Artificial Intelligence for disease diagnosis. Biochips & their applications. Biosensors & their applications.

Reference Books:

1. Rajendra Singh C and Rathnakar Rao N, Rajendra Singh C., Rathnakar Rao, "Biology for Engineers" N. Publishing, Bengaluru, 2023.
2. S., Selvamurugan N., Rajesh M.P., Nazeer R.A., Thilagaraj W., Barathi S., and Jaganthan M.K., *Biology for Engineers*, Tata McGraw-Hill, New Delhi, 2012.
3. Arthur T. Johnson, *Biology for Engineers*, 2nd Edition, CRC Press, Taylor & Francis Group, Boca Raton, FL, 2019.
4. Sohini Singh and Tanu Allen, Biology for Engineers, Vayu Education of India, New Delhi, 2014.
5. Yoseph Bar-Cohen, Biomimetics: Nature-Based Innovation, 1st edition, CRC Press., 2012
6. Ibrahim Ozbolat, 3D Bioprinting: Fundamentals, Principles and Applications, Academic Press (Elsevier), 2016.
7. Rodríguez Méndez, M. L. (Ed.), Electronic Noses a*nd Tongues in Food Science***,** 1st Edition, Academic Press (Elsevier), Cambridge, MA, USA, 2016**.**

Web links and Video Lectures (e-Resources)

- https://nptel.ac.in/courses/121106008
- https://freevideolectures.com/course/4877/nptel-biology-engineers-other-non-biologists
- https://ocw.mit.edu/courses/20-020-introduction-to-biological-engineering-design-spring-2009
- https://ocw.mit.edu/courses/20-010j-introduction-to-bioengineering-be-010j-spring-2006
- https://www.coursera.org/courses?query=biology
- https://onlinecourses.nptel.ac.in/noc19_ge31/preview
- https://www.classcentral.com/subject/biology
- https://www.futurelearn.com/courses/biology-basic-concepts

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO 1	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	3 (WK5)	2 (WK8)	2 (WK2, WK6)	2 (WK1, WK5, WK7)	2 (WK9)	2	–	–	3 (WK8)	3	2	2
CO 2	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	3 (WK1, WK5,WK7)	2 (WK9)	2	–	–	3 (WK8)	3	2	2
CO 3	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	3 (WK1, WK5,WK7)	3 (WK9)	2	–	–	3 (WK8)	3	2	2
CO 4	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	3 (WK1, WK5,WK7)	3 (WK9)	2	–	–	3 (WK8)	3	2	2

Course Code		:	1BHSA424C	Universal Human Values-II	Semester	:	04
L:T:P		:	1:0:0		Course Type	:	UHV
Hours/Sem.	Teaching	:	14Hrs		CIE Marks	:	50
	Learning (TW+SL)	:	13Hrs		SEE Marks	:	50
	Exam	:	3Hrs		Total Marks	:	100
	Total Hrs.	:	30Hrs		Credit	:	01

Professional Competency:

After completing the course, students will be able to understand and apply value-based principles to achieve personal harmony, nurture ethical relationships, and promote sustainable and responsible living in personal, social, and professional life.

Course Outcomes:

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

1. Understand the need and role of value education for holistic development
2. Differentiate between self and body, needs, and practice harmony within the human being.
3. Analyze relationship dynamics based on trust, respect, and justice within families and society.
4. Recognize harmony in nature and existence, and explore ethical practices and professional conduct within personal and professional contexts.

Unit-I	3Hrs.

Introduction to Value Education

Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfill the Basic Human Aspirations

Unit-II	4Hrs.

Harmony in the Human Being :

Understanding Human being as the Co-existence of the Self and the Body, Distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self, Understanding Harmony in the Self, Harmony of the Self with the Body, Program to ensure self-regulation and Health

Unit-III	3Hrs.

Harmony in the Family and Society :

Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Valuein Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in Human-to-Human Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order

Unit-IV	4Hrs.

Harmony in the Nature/Existence Implications of the Holistic Understanding – a Look at Professional Ethics :

Understanding Harmony in the Nature, Interconnectedness, self-regulation and Mutual Fulfillment among the Four Orders of Nature, Realizing Existence as Co-existence at All Levels, The Holistic Perception of Harmony in Existence.

Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics Holistic Technologies, Production Systems and Management Models-Typical Case Studies, Strategies for Transition towards Value-based Life and Profession.

Reference Books:

1. Gaur R R, Asthana,R.G, Bagaria G P (2019).A Foundation Course in Human Values and Professional Ethics,2nd Revised Edition, Excel Books, New Delhi,
2. The Teacher‟s Manual for A Foundation Course in Human Values and ProfessionalEthics, R R Gaur, R Asthana, G

3. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amar kantak, 1999.
4. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
5. The Story of Stuff (Book).
6. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
7. Small is Beautiful - E. F Schumacher.
8. Andrews Cecile, Slow is Beautiful
9. Kumarappa J C ,Economy of Permanence
10. Pandit Sunderlal ,Bharat Mein Angreji Raj
11. Dharampa l, Rediscovering India
12. Gandhi Mohandas K ,Hind Swaraj or Indian Home Rule.
13. Maulana Abdul Kalam Azad :India Wins Freedom -
14. Romain Rolland (English) Vivekananda
15. Romain Rolland (English), Gandhi
16. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986, 1991
17. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome's report, Universe Books.
18. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
19. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
20. A N Tripathy, 2003, Human Values, New Age International Publishers.
21. SubhasPalekar, 2000, How to practiceNatural Farming, Pracheen (Vaidik) KrishiTantraShodh, Amravati.
22. E G Seebauer & Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers , Oxford University Press
23. M Govindrajran, S Natrajan & V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.
24. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
25. B L Bajpai, 2004, Indian Ethos and Modern Management, New Royal Book Co., Lucknow. Reprinted 2008.

Web Links and Video Lectures:

Suggested Web Links:

1. Value Education websites,
2. https://www.uhv.org.in/uhv-ii,
3. http://uhv.ac.in,
4. http://www.uptu.ac.in
5. Story of Stuff,
6. http://www.storyofstuff.com
7. Al Gore, An Inconvenient Truth, Paramount Classics, USA
8. Charlie Chaplin, Modern Times, United Artists, USA
9. IIT Delhi, Modern Technology – the Untold Story
10. Gandhi A., Right Here Right Now, Cyclewala Productions
11. https://www.youtube.com/channel/UCQxWr5QB_eZUnwxSwxXEkQw
12. https://fdp-si.aicte-india.org/8dayUHV_download.php
13. https://www.youtube.com/watch?v=8ovkLRYXIjE
14. https://www.youtube.com/watch?v=OgdNx0X923I
15. https://www.youtube.com/watch?v=nGRcbRpvGoU
16. https://www.youtube.com/watch?v=sDxGXOgYEKM

CO PO Mapping

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	–	–	–	–	–		2 (WK9)	–	–	–	3 (WK8)	1	–	–
CO2	–	–	–	–	–	2 (WK5, WK7)	–	–	–	–	3 (WK8)	1	–	–
CO3	–	–	–	–	2 (WK5, WK7)	–	–	–	–	3 (WK8)	2 (WK8)	1	–	–
CO4	–	–	–	–	2 (WK5, WK7)	–	–	–	–	3 (WK8)	2 (WK8)	1	–	–

Fifth Semester Syllabus

Course Code		**:**	**1BUEA501C**	**Data Science and Visualization**	**Semester**	**:**	**05**
L:T:P		**:**	**3:0:2**		**Course Type**	**:**	**IPCC**
Hours/Sem.	**Teaching**	**:**	**42+26=68 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**44 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Analyze, visualize, and interpret data using modern data science techniques and visualization tools to solve engineering and real-world problems and support informed decision-making.

Course Outcomes:

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

CO1: Apply different techniques to Explore Data Analysis and the Data Science Process

CO2: Analyze feature selection algorithms & design a recommender system.

CO3: Evaluate data visualization tools and libraries and plot graphs.

CO4: Develop different charts and include mathematical expressions

Unit-I	**14 Hrs.**

Introduction to Data Science: What is Data Science? Big Data and Data Science hype – and getting past the hype, Why now? – Datafication, Current landscape of perspectives, Skill sets. Needed Statistical Inference: Populations and samples, Statistical modelling, probability distributions, fitting a model. **Exploratory Data Analysis and the Data ScienceProcess:** Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process.

Unit-II	**14 Hrs.**

Case Study: Real Direct (online real estate firm), **Feature Generation and Feature Selection:** Extracting Meaning from Data Motivating application: user (customer) retention. Feature Generation (brainstorming, role of domain expertise, and place for imagination), Feature Selection algorithms. Filters; Wrappers; Decision Trees; Random Forests. **Recommendation Systems:** Building a User-Facing Data Product, Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, **Exercise:** build your own recommendation system.

Unit-III	**14 Hrs.**

Data Visualization and Data Exploration: Data Visualization, Importance of Data Visualization, Data Wrangling, Tools and Libraries, for Visualization. **Comparison Plots:** Line Chart, Bar Chart and Radar Chart; **Relation Plots:** Scatter Plot, Bubble Plot , Correlogram and Heatmap; **Composition Plots:** Pie Chart, Stacked Bar Chart, Stacked Area Chart, Venn Diagram; **Distribution Plots:** Histogram, Density Plot, Box Plot, Violin Plot; **Geo Plots:** Dot Map, Choropleth Map, Connection Map; What Makes a Good Visualization?

Unit-IV	**14 Hrs.**

A Deep Dive into Matplotlib: Introduction, Overview of Plots in Matplotlib, **Pyplot Basics:** Creating Figures, Closing Figures, Format, Strings, Plotting, Plotting Using pandas DataFrames, Displaying Figures, Saving Figures; **Basic Text and Legend Functions:** Labels, Titles, Text, Annotations, Legends; Basic

Plots:Bar Chart, Pie Chart, Stacked Bar Chart, Stacked Area Chart, Histogram, Box Plot, Scatter Plot, Bubble Plot; **Layouts:** Subplots, Tight Layout, Radar Charts, GridSpec; **Images:** Basic Image Operations, Writing Mathematical Expressions

Reference Books:

1. Cathy O'Neil and Rachel Schutt, "Doing Data Science", 2013, O'Reilly Media, Inc O'Reilly Media, Inc
2. Tim Grobmann and Mario Dobler, "Data Visualization workshop", Packt Publishing, ISBN 9781800568112
3. Anand Rajaraman and Jeffrey D. Ullman, "Mining of Massive Datasets", 2010, CambridgeUniversity Press
4. Joel Grus, "Data Science from Scratch", Shroff Publisher /O'Reilly Publisher Media
5. Andy Krik, "A Handbook for Data Driven Design"

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	2	–
CO2	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6, WK8)	3	3	1
CO3	2 (WK1, WK2, WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6)	3	2	1
CO4	3 (WK1, WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	2

Course Code		:	**1BUEA502C**	**Digital System Design using Verilog**	**Semester**	:	**05**
L:T:P		:	**4:0:0**		**Course Type**	:	**PCC**
Hours/Sem.	**Teaching**	:	**56 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**58 Hrs.**		**SEE Marks**	:	**50**
	Exam	:	**06 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**120 Hrs.**		**Credits**	:	**04**

Professional Competency:

Design, model, simulate, and verify digital systems using Verilog Hardware Description Language (HDL) by applying appropriate modeling techniques and modern Electronic Design Automation (EDA) tools to develop reliable and efficient digital hardware solutions.

Course Outcomes:

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

- **CO1:** Explain the Verilog HDL design flow, language constructs, and modeling concepts used in digital system design. .
- **CO2:** Develop and simulate Verilog HDL programs using Gate-level and Dataflow modeling techniques for combinational and sequential digital circuits.
- **CO3:** Design and implement digital systems using Behavioral and Structural modeling approaches in Verilog HDL.
- **CO4:** Analyze and optimize Verilog HDL programs by effectively utilizing Tasks and Functions to improve modularity, reusability, and code efficiency.

Unit-I	**14 Hrs.**

Overview of Digital Design with Verilog HDL: Evolution of Computer-Aided Digital Design (CAD), Emergence of HDLs, Typical Design flow, Importance of HDLs, Popularity of Verilog HDL, Trends in HDLs. **Hierarchical Modeling Concepts**: Design Methodologies, Top-down and Bottom-up design methodology, Modules, Instances, Components of a Simulation, Design Block, Stimulus Block (Test Bench) with example.

Unit-II	**14 Hrs.**

Basic Concepts: Lexical Conventions, Data Types, System Tasks, Compiler Directives.**Modules and Ports:** Modules, Ports, Connecting Ports, Hierarchical Names. **Gate-Level Modeling**: Gate Types-Modeling using basic Verilog gate primitives, Description of AND/OR and BUF/NOT type gates. Gate Delays-Rise, Fall and Turn-Off Delays, Min, Max and Typical Delays.

Unit-III	**14 Hrs.**

Dataflow Modeling: Continuous assignments, Delay Specification, Expressions, Operators, Operands, Operator Types, Examples. **Behavioral Description**: Structured Procedures, Initial and Always statements, Procedural Assignments Blocking and Non-Blocking statements, Conditional statements, Multiway Branching, Loops, Sequential and Parallel blocks, Examples.

Unit-IV	**14 Hrs.**

Structural Description: Highlights of Structural Descriptions, Organization of Structural Description, Binding. **Tasks and Functions**: Differences between Tasks and Functions, Declaration and Invocation, Examples.

Reference Books:

1. Samir Palnitkar, "Verilog HDL: A Guide to Digital Design and Synthesis", 2nd Edition, Pearson Education.
2. Nazeih M Botros, "HDL programming (VHDL and Verilog)", 2008, John Wiley India Pvt. Ltd.
3. Donald E. Thomas, Philip R. Moorby, "The Verilog Hardware Description Language", 5th Edition, Springer Science + Business Media, LLC.
4. Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL" , 2nd Edition, Pearson (Prentice Hall).
5. Padmanabhan, Tripura Sundari, "Design through Verilog HDL", Wiley, 2016, or earlier.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	–	2 (WK2, WK6)	–	–	–	–	–	1 (WK4)	3	2	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	3	3
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		**:**	**1BUEA503L**	**Verilog Laboratory**	**Semester**	**:**	**05**
L:T:P		**:**	**0:0:2**		**Course Type**	**:**	**PCCL**
Hours/Sem.	**Teaching**	**:**	**26 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**--**		**SEE Marks**	**:**	**50**
	Exam	**:**	**04 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**30 Hrs.**		**Credits**	**:**	**01**

Professional Competency:

Design, implement, simulate, and verify digital systems using Verilog HDL and modern Electronic Design Automation (EDA) tools by applying appropriate hardware modeling techniques to develop reliable and efficient digital solutions.

Course Outcomes:

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

CO1: Exhibit the Verilog Programming skills and model the digital system at different abstraction levels.

CO2: Write Test bench to verify the design.

CO3: Interface various input and output devices to FPGA.

CO4: Develop a complete system for given specifications.

List of Experiments

1. Write Verilog program for the following combinational design along with test bench to verify the design:
   A. 2 to 4 decoder realization using NAND gates only (structural model).
   B. 8 to 3 encoder with priority and without priority (behavioural model).
   C. 8 to 1 multiplexer using case statement and if statements.
   D. 4-bit binary to gray converter using 1-bit gray to binary converter 1-bit adder and subtractor.
2. Model in Verilog for a full adder and add functionality to perform logical operations of XOR, XNOR, AND and OR gates. Write test bench with appropriate input patterns to verify the modeled behaviour.
3. Write Verilog code for SR, D and JK and verify the flip flop.
4. Write Verilog code for 4-bit BCD synchronous counter.
5. Write Verilog code for counter with given input clock and check whether it works as clock divider performing division of clock by 2, 4, 8 and 16. Verify the functionality of the code.
6. Write a Verilog code to design a clock divider circuit that generates 1/2 and 1/4th clock from a given input clock. Port the design to FPGA and validate the functionality through oscilloscope.
7. Interface a DC motor to FPGA and write Verilog code to change its speed and direction.
8. Interface a Stepper motor to FPGA and write Verilog code to control the Stepper motor rotation.
9. Interface a DAC to FPGA and write Verilog code to generate Sine wave of frequency F KHz (eg. 200 KHz) frequency.
10. Write Verilog code to interface 7-segment display digits/LCD display.
11. Write Verilog code using FSM to simulate elevator operation.
12. Write Verilog code to interface simple Switches and LEDs.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK5)	1 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	3	3
CO2	2 (WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6, WK8)	3	3	2
CO3	3 (WK2, WK3, WK4)	2 (WK2, WK3)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		**:**	**1BUEA504E**	**IoT and Applications**	**Semester**	**:**	**05**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop IoT applications using microcontrollers, cloud platforms, and edge computing technologies for real-time monitoring and control.

Course Outcomes:

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

CO1: Explain the evolution of IoT, IoT networking components, and addressing strategies in IoT.

CO2: Analyze various sensing devices and actuator types and demonstrate the processing in IoT.

CO3: Apply different connectivity technologies.

CO4: Elaborate the need for data analytics and security in IoT.

Unit-I	**10 Hrs.**

Introduction to Internet of Things: Introduction, Physical design of IoT, Logical Design of IoT, IoT enabling technologies, IoT Levels and Deployment Templates. IoT and M2M: Introduction: M2M, Difference between IoT and M2M, SDN and NFV for IoT, IoT System Management with NETCONF-YANG, Need for IoT Systems Management.

Unit-II	**11 Hrs.**

AVR Architecture and Assembly Language Programming: The General Purpose Registers in the AVR, The AVR Data Memory, Using Instructions with the Data Memory, AVR Status Register, AVR Data Format and Directives, Introduction to AVR Assembly Programming, The Program Counter and Program ROM Space in the AVR, RISC Architecture in the AVR, **AVR I/O Port Programming:** I/O Port Programming in AVR, I/O Bit Manipulation Pragramming.

Unit-III	**11 Hrs.**

Arithmetic, Logical instructions and Programs: Arithmetic Instructions, Signed Number Concepts and Arithmetic Operations, Logical and Compare Instructions, Rotate and Shift Instructions and Data Serialization, BCD and ASCII Conversion. Branch, **Call and Time Delay Loops:** Branch Instructions and Looping, Call Instructions and Stack, AVR Time Delay and Instruction pipeline.

Unit-IV	**10 Hrs.**

Case Studiesillustrating IoT design – Home Automation, Cities, Agriculture Data Analytics for IoT: Introduction, Apache Hadoop, Using Hadoop MapReduce for Batch Data Analytics, Apache Oozie, Apache Spark, Apache Storm, Using Apache Storm for Real-time Data Analysis.

Reference Books:

1. Arshdeep Bahga, Vijay Madisetti, “Internet of Things- A Hands-On Approach”, 2014, Universities Press.
2. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, 1st

Edition, Pearson Education (Cisco Press Indian Reprint). (ISBN: 978-9386873743).
3. Srinivasa K G, “Internet of Things”, 2017, Cengage Leaning India.
4. Muhammad Ali Mazidi, Sarmad Naimi, Sepehr Naimi, “The AVR Microcontroller and Embedded Systems: Using Assembly and C”, 2014, Pearson Education Limited

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK5)	1 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6)	3	3	3
CO2	3 (WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK6, WK8)	3	3	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	1 (WK8)	1 (WK8)	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		**:**	**1BUEB504E**	**Principles of Communication Systems**	**Semester**	**:**	**05**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Apply the fundamental principles of communication systems to analyze analog and digital communication techniques for reliable information transmission

Course Outcomes:

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

CO1 : Explain the concepts of probability, random variables, and random processes used in communication systems.

CO2 : Analyze analog modulation and demodulation techniques including AM, FM, DSB-SC, SSB, and VSB.

CO3 : Demonstrate the digitization of analog signals using sampling, pulse modulation, PCM, and line coding techniques.

CO4 : Evaluate the performance of digital modulation techniques such as ASK, PSK, FSK, MSK, and DPSK using error probability concepts.

Unit-I	**10 Hrs.**

Random Variables and Processes: Introduction, probability, conditional probability. Random variables: Continuous and discrete random variable, statistical averages, distribution and density functions, central limit theorem. **Random processes**: Mean, correlation and covariance function, properties of autocorrelation function, cross–correlation function, Gaussian process.

Unit-II	**10 Hrs.**

Amplitude Modulation (AM) Fundamentals: AM concepts, time domain and frequency domain description, generation and detection of AM waves, Generation of DSB-SC, SSB, VSB modulation, super heterodyne receiver. **Angle modulation:** Concept of angle modulation, relation between frequency and phase modulation, bandwidth of angle modulated wave. Generation of FM: Direct and indirect methods, PLL, demodulation of FM, pre-emphasis and de-emphasis, FM radio.

Unit-III	**11 Hrs.**

Digital Representation of Analog Signals: Introduction, sampling process, pulse amplitude modulation, time-division multiplexing. **Pulse Position Modulation (PPM):** Generation and detection of PPM wave, the quantization process. **Pulse Code Modulation**: Sampling, quantization, encoding, line codes, differential encoding.

Unit-IV	**11 Hrs.**

Digital Modulation Techniques: Digital Modulation formats, Coherent binary modulation techniques (ASK, PSK, FSK), Probability of error for each ASK, PSK, FSK. **Coherent quadrature modulation techniques**: Minimum Shift Keying. **Non-coherent binary modulation techniques:** Frequency Shift Keying and Differential Phase Shift Keying

Reference Books:

1. Louis E Frenzel, Principles of Electronic Communication Systems, Mc Graw Hill Education (India) Private Limited, 3rd Edition, 2016. ISBN: 978-0-07-066755-6.
2. Simon Haykin, Michael Moher, "Communication Systems", John Wiley, India Pvt. Ltd, 5th Edition, 2010, ISBN:978-81-265-2151-7
3. B. P. Lathi, Zhi Ding, "Modern Digital and Analog Communication Systems", Oxford University Press., 4th edition, 2010, ISBN: 97801980738002.
4. Herbert Taub, Donald L. Schilling, Goutam Saha, "Principles of Communication systems", Mc Graw Hill Education (India) Private Limited, 4th Edition, 2016. ISBN: 978-1-25-902985-1
5. John G. Proakis, Masoulsalehi, "Fundamental of Communication System",2014, Pearson Education.
6. Sam Shanmugan K. (2006), "Digital and Analog Communication Systems", John Wiley & Son.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	1 (WK3, WK4)	–	–	2 (WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	3	1
CO3	2 (WK3, WK4)	3 (WK3, WK4)	2 (WK5)	1 (WK8)	2 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	2
CO4	2 (WK3, WK4)	2 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3

Course Code		:	**1BUEC504E**	**Operating Systems**	**Semester**	:	**05**
L:T:P		:	**3:0:0**		**Course Type**	:	**PEC**
Hours/Sem.	**Teaching**	:	**42 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**42 Hrs.**		**SEE Marks**	:	**50**
	Exam	:	**06 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**90 Hrs.**		**Credits**	:	**03**

Professional Competency:

Apply concepts of process management, CPU scheduling, synchronization, deadlock handling, and inter-process communication to solve system-level problems

Course Outcomes:

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

CO1: Identify the structure of an operating system and its scheduling mechanism.

CO2: Demonstrate the allocation of resources for a process using scheduling algorithm.

CO3: Identify root causes of deadlock and provide the solution for deadlock elimination.

CO4: Explore about the storage structures and learn about the Linux Operating system.

Unit-I	**10 Hrs.**

Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments. **Operating System Services:** User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot, **Process Management:** Process concept; Process scheduling; Operations on processes; Inter process communication.

Unit-II	**11 Hrs.**

Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling. **Process Synchronization:** Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors, **Deadlocks:** Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Unit-III	**10 Hrs.**

Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation, **Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System, Implementation of File System:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.

Unit-IV	**11 Hrs.**

Secondary Storage Structures, Protection: Mass storage structures; Disk structure; Disk attachment; Disk

scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems, **Case Study:** The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication.

Reference Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "Operating System Principles", 7th Edition, Wiley-India, 2006.
2. Ann McHoes Ida M Fylnn, "Understanding Operating System", Cengage Learning, 6th Edition.
3. D.M Dhamdhere, "Operating Systems: A Concept Based Approach", 3rd Edition, McGraw- Hill, 2013.
4. P.C.P. Bhatt, "An Introduction to Operating Systems: Concepts and Practice", 4th Edition, PHI (EEE), 2014.
5. William Stallings, "Operating Systems: Internals and Design Principles", 6th Edition, Pearson.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	2 (WK3, WK4)	–	–	–	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	2 (WK3, WK4)	–	–	–	–	–	–	–	–	–	3	3	1
CO3	2 (WK3, WK4)	3 (WK3, WK4)	–	–	–	–	–	–	–	–	–	3	3	2
CO4	2 (WK3, WK4)	2 (WK3, WK4)	–	–	–	–	–	–	–	–	–	3	3	3

Course Code		**:**	**1BUED504E**	**Finite Automata and Formal Languages**	**Semester**	**:**	**05**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Apply the principles of finite automata, regular expressions, context-free grammars, and formal languages to model computational systems.

Course Outcomes:

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

CO1: Understand the fundamentals of finite automata theory and regular expressions.

CO2: Demonstrate the properties of regular languages and context free grammars and languages.

CO3: Demonstrate the properties of context-free Languages and push-down automata.

CO4: Exhibit the fundamentals and programming techniques of turing machines and un-decidability concepts.

Unit-I **10 Hrs.**

Introduction to Finite Automata: Introduction to Finite Automata; The central concepts of automata theory; Deterministic finite automata; Nondeterministic finite automata. **Finite Automata, Regular Expressions:** An application of finite automata; Finite automata with Epsilon-transitions; Regular expressions; Finite automata and regular expressions; Applications of regular expressions.

Unit-II **11 Hrs.**

Regular Languages, Properties of Regular Languages: Regular languages; Proving languages not to be regular languages; Closure properties of regular languages; Decision properties of regular languages; Equivalence and minimization of automata. **Context-Free Grammars and Languages:** Context –free grammars, Parse trees, Applications, Ambiguity in grammars and Languages.

Unit-III **11 Hrs.**

Pushdown Automata: Definition of the pushdown automata, The languages of a PDA, Equivalence of PDAs and CFGs, Deterministic pushdown automata. **Properties of Context-Free Languages:** Normal forms for CFGs, The pumping lemma for CFGs, Closure properties of CFLs.

Unit-IV **10 Hrs.**

Introduction To Turing Machine: Problems that computer cannot solve, The turning machine, Programming techniques for turing machines, 50 Extensions to the basic turing machines, Turing machine and computers. **Un-decidability:** A language that is not recursively enumerable, An un-decidable problem that is RE, Post's correspondence problem, Other un-decidable problems.

Reference Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, 2007, Pearson Education.
2. K.L.P. Mishra, "Theory of Computer Science, Automata, Languages, and Computation", 3rd Edition, 2009, PHI Learning.

3. Raymond Greenlaw, H.James Hoover, "Fundamentals of the Theory of Computation Principles and Practice", 1998, Elsevier.
1. John C Martin, "Introduction to Languages and Automata Theory", 3rd Edition, 2007, Tata McGraw-Hill.
2. Thomas A. Sudkamp, "An Introduction to the Theory of Computer Science, Languages and Machines", 3rd Edition, 2006, Pearson Education.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK3, WK4)	2 (WK2, WK3, WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK2, WK3, WK5)	2 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	1
CO3	2 (WK3, WK4)	3 (WK3, WK4)	3 (WK2, WK3, WK5)	2 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	2
CO4	2 (WK3, WK4)	3 (WK3, WK4)	3 (WK2, WK3, WK5)	2 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	3

Course Code		**:**	**1BUEA531N**	**Electrical Sensors and Transducers**	**Semester**	**:**	**05**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**OEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Understand the operating principles, characteristics, and performance parameters of electrical sensors and transducers used in measurement and control systems.

Course Outcomes:

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

CO1: Describe the characteristics of various sensors/transducers.

CO2: Interpret the construction and working of various sensors/transducers.

CO3: Select an appropriate sensor for real time measurement application.

CO4: Use an appropriate sensor for designing an engineering solution for real time measurement of process parameter.

Unit-I	**11 Hrs.**

Introduction to Sensor: Meaning of sensors and transducers, classification of transducers (Mechanical/electrical, Active/passive, Absolute/Modulated output type), Sensor based measurement system, Static characteristics (Accuracy, precision, sensitivity, error, linearity, resolution, hysteresis), Dynamic characteristics (Order and step response of I, II order system) **Resistive Sensors:** Potentiometers, Strain gauges, Resistive temperature detectors (RTD), Thermistors, Magnetoresistors, Light-dependent resistors (LDR), Resistive hygrometer, Resistive gas sensor, Liquid conductivity sensors.

Unit-II	**11 Hrs.**

Self-Generating Sensors: Thermoelectric sensors: Thermocouples, Piezoelectric sensors, Hall effect sensors, Pyroelectric sensors, Photo voltaic sensors, Electrochemical sensors. **Capacitive Sensors:** Variable capacitors: Differential arrangement, Different applications. Differential capacitor: Principle based on variable are, distance and dielectric property, applications.

Unit-III	**10 Hrs.**

Inductive Sensors: Variable reluctance type, Eddy current type, LVDT, LVDT applications, Variable transformers: Synchros, Resolvers, Inductosyns. Magnetoelastic and magnetostrictive sensors, Super Quantum Interference Devices (SQUIDS). **Electromagnetic Sensors:** Sensors based on Faraday's law, Linear velocity sensor, Search coil magnetometer, Electromagnetic flow meter.

Unit-IV	**10 Hrs.**

Digital and Intelligent Sensors: Position encoders, Resonant sensors, Intelligent sensors. **Miscellaneous Sensors:** Sensors based on semiconductor junctions, Sensors based on MOSFET transistors, Fiber optic sensors, Ultrasonic based sensors, Biosensors.

Reference Books:

1. Ramon P. Areny, John G. Webster, "Sensors and Signal Conditioning", 2nd Edition, Wiley India Private Ltd.

2. Ian R. Sinclair, "Sensors and Transducers", 3rd Edition, Newnes Publication.
3. D. Patranabis, "Sensors and Transducers", 2nd Edition, PHI.
4. Allan S. Morris, "Measurement and Instrumentation Principles", 3rd Edition, Butterworth and Heinmann Publication.
5. John P. Bentley, "Principles of Measurement Systems", 3rd Edition, 2004, Pearson Publication.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	2	1	1
CO2	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3, WK4, WK6)	2 (WK2, WK3, WK4, WK6)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	2	1	1
CO3	3 (WK1, WK2, WK3, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	2	1	1
CO4	3 (WK1, WK2, WK4)	2 (WK2, WK3, WK4, WK6)	3 (WK2, WK3, WK4, WK6)	3 (WK3, WK4, WK6)	2 (WK2, WK6)	–	–	–	1 (WK8)	–	2 (WK8)	2	1	1

Course Code		:	1BHSA522C	Quantitative Aptitude and Professional Skills	Semester	:	05
L:T:P		:	2:0:0		Course Type	:	AEC
Hours/Sem.	Teaching	:	28Hrs		CIE Marks	:	50
	Learning (TW+SL)	:	28Hrs		SEE Marks	:	50
	Exam	:	04Hrs		Total Marks	:	100
	Total Hrs.	:	60Hrs		Credit	:	00

Professional Competency:

Demonstrate competency in enhancing vocabulary, improving verbal ability, analyzing problems systematically, and solving quantitative problems using appropriate language and numerical techniques.

Course Out comes :

At the end of the course student will be able to

CO1: Enhance vocabulary and apply techniques to further improve word usage and language proficiency.

CO2: Apply techniques to improve verbal ability in communication and comprehension tasks.

CO3: Analyze problems step-by-step and develop appropriate methods for solving them effectively.

CO4: Solve quantitative problems by applying appropriate concepts and techniques to improve numerical ability.

Unit-I	**7Hrs.**

Vocabulary Development: Vocabulary Building Techniques, Root Words, Antonyms & Synonyms, Sentence Completion, Error Detection & Correction, Reading Comprehension

Unit-II	**7Hrs.**

Numbers, Proportion & Finance: Number System, Factors & Multiples, The God of Math–Linear Equations, Ratio-Proportion-Variation, Percentages, Profit & Loss, Interest, Averages & Allegations

Unit-III	**7Hrs.**

Time & Probability: Time & Work, Time, Speed & Distance, Permutations & Combinations, Probability

Unit-IV	**7Hrs.**

Verbal, Analytical, and Visual Reasoning: Human Relations, Direction Tests, Coding Decoding, Clocks and Calendars, Visual Reasoning, Analytical Puzzles, Mathematical, Arrangement & Classification Puzzles

Reference Books:

1. R.S. Aggarwal, “A Modern Approach to Verbal and Non–Verbal Reasoning”, Sultan Chand and Sons, New Delhi, 2018
2. R. S. Aggarwal, “Quantitative Aptitude”, Sultan Chand and Sons, New Delhi, 2018
3. Chopra, “Verbal and Non–Verbal Reasoning”, MacMillan India
4. M Tyra, “Magical Book on Quicker Maths”,BSCPublications,2018
5. George J Summers, “The Great Book of Puzzles &T easers”, Jaico Publishing House, 1989
6. ShakuntalaDevi,“PuzzlestoPuzzleYou”,OrientPaperBacks,NewDelhi,1976
7. R.S.Aggarwal,“AModernApproachtoLogicalReasoning”,SultanChandandSons,New Delhi, 2018
8. Cambridge Advanced Learner’s Dictionary, Cambridge University Press.

Kaplan’s GRE guide

CO-PO Mapping

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	–	2 (WK3, WK4)	–	–	–	–	–	–	–	2 (WK8)	–	3	2	–
CO2	–	3 (WK3, WK4)	–	–	–	–	–	–	–	2 (WK8)	–	3	3	1
CO3	–	–	3 (WK2, WK3, WK5)	–	2 (WK2, WK6)	–	–	–	–	–	–	3	3	2
CO4	–	–	3 (WK2, WK3, WK5)	–	2 (WK2, WK6)	–	–	–	–	–	–	3	3	3

Course Code		:	1BBTA523C	Environmental Studies	Semester	:	05
L:T:P		:	1:0:0		Course Type	:	MC
Hours/Sem.	Teaching	:	14 Hrs.		CIE Marks	:	50
	Learning (TW+SL)	:	13 Hrs.		SEE Marks	:	50
	Exam	:	03 Hr.		Total Marks	:	100
	Total Hrs.	:	30 Hrs.		Credits	:	01

Professional Competency:

Develop the ability to analyze environmental issues and implement sustainable solutions by applying environmental science principles, pollution control measures, and environmental legislation.

Course Outcomes:

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

CO1: Identify natural resources and its uses.
CO2: Understand pollution and its effects on environment and to implement sustainable future in the work place.
CO3: Analyze current environmental issues.
CO4: Apply the waste management techniques in various fields.

UNIT – I	03 Hrs.

Natural Resources:

Human activities and their impacts. Environmental Impact Assessment, **Renewable Energy**: Solar, Wind, Hydropower, Tidal, Ocean thermal, Geo thermal, Biomass energy, Biogas, Biodiesel, Bioethanol, Hydrogen as fuel. **Non renewable Energy**: Coal, Petroleum, Natural gas, Nuclear energy.

UNIT – II	04 Hrs.

Environmental Pollution: Water pollution, water quality standards, water borne diseases, Fluoride problem, Air pollution, Noise pollution. Effect of electromagnetic waves. **Sustainable future**: Concept of sustainable development, threats to sustainability, strategies for sustainable development, Sustainable development goals (SDGs). Environment economics – concept of green building, Circular Economy.

UNIT – III	03 Hrs.

Current Environmental Issues of concern: Greenhouse Effect- Greenhouse gases and Global Warming, Climate change, ozone layer depletion, Acid rain, Eutrophication, Environmental policy legislation rules & regulations

UNIT – IV	04Hrs.

Fundamentals of Waste management:

Solid waste management: Sources, classification, characteristics, collection & transportation, disposal, and processing methods. Hazardous waste management and handling.
Concept of waste water treatment, Bioremediation.
Industrial waste management (Case studies: Cement, plastic, chemical, E–waste, food & construction industry waste management).

Text books

1. Benny Joseph, "Environmental Studies", 3rd Edition, McGraw Hill Education (India) Pvt. Ltd., 2017.
2. Manjunath, D. L., "Environmental Studies", 1st Edition, Pearson Education, New Delhi, 2007.
3. Anubha Kaushik and C. P. Kaushik, "Perspectives in Environmental Studies", 8th Edition, New Age International (P) Ltd., New Delhi, 2025.
4. P. Meenakshi, "Elements of Environmental Science and Engineering", 2nd Edition, PHI Learning Pvt. Ltd., New Delhi, 2012.

CO-PO Mapping

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	2 (WK1, WK2, WK3, WK4)	–	2 (WK2, WK3, WK5)	–	–	3 (WK1, WK5, WK7)	–	–	–	–	3 (WK8)	3	–	–
CO2	2 (WK1, WK2, WK3, WK4)	–	2 (WK2, WK3, WK5)	–	–	3 (WK1, WK5, WK7)	–	–	–	–	3 (WK8)	3	–	1
CO3	2 (WK1, WK2, WK3, WK4)	–	3 (WK2, WK3, WK5)	–	–	3 (WK1, WK5, WK7)	–	–	–	–	3 (WK8)	3	–	2

Sixth Semester Syllabus

Course Code		**:**	**1BUEA601C**	**Machine Learning**	**Semester**	**:**	**06**
L:T:P		**:**	**3:0:2**		**Course Type**	**:**	**IPCC**
Hours/Sem.	**Teaching**	**:**	**42+26=68 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**44Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120 Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Design, analyze, and implement suitable Machine Learning models and algorithms to extract insights from data and develop intelligent systems for real-world applications.

Course Outcomes:

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

CO1: Explain the fundamental concepts, need, and applications of Machine Learning and its relation to other disciplines.

CO2: Apply statistical and analytical techniques for understanding, preparing, and visualizing data in Machine Learning.

CO3: Demonstrate concept learning, hypothesis representation, and the use of basic learning algorithms such as Find-S.

CO4: Implement and analyze similarity-based, regression, decision tree, clustering, and Bayesian learning algorithms for predictive modeling.

UNIT–I **10 Hrs**

Need for Machine Learning, Why Machine Learning is popular? What is Machine Learning? Knowledge Pyramid, Machine Learning in relation to other fields, Types of Machine Learning, Challenges of Machine Learning, Machine Learning Process, Machine Learning Application.**Understanding Data**: Introduction, Big Data Analytics and Types of Analytics, Big Data Analysis Framework.

UNIT–II **11 Hrs**

Descriptive Statistics: Introduction: Why Descriptive Statistics in ML? Dataset and Data Types, Univariate Data Analysis & Visualization, Bivariate Data and Multivariate Data. Multivariate Data: What is Multivariate Data? Essential Mathematics for Multivariate Data, Machine Learning and Importance of Probability and Statistics.**Overview of Hypothesis**: Comparing Learning Methods. Introduction to Learning and its types, Designing of Learning System, Concept Learning, Representation of a Hypothesis, Hypothesis space search by Find-S algorithm.

UNIT-III **11 Hrs**

Similarity-based Learning: Introduction to Similarity or Instance-based Learning, Nearest-Neighbor Learning, Weighted K-Nearest-Neighbor Algorithm, Nearest Centroid Classifier, Locally Weighted Regression (LWR). Regression Analysis: Introduction to Regression, Introduction to Linearity, Correlation, and Causation, Introduction to Linear Regression, Validation of Regression Methods, Multiple Linear Regression, Polynomial Regression, Logistic.

UNIT-IV	10 Hrs

Decision Tree Learning: Introduction to Decision Tree Learning Model, Decision Tree Induction Algorithms: ID3 and C4.5, Regression Trees, Validating and Pruning of Decision Trees. **Clustering**: K-Means Clustering. Bayesian Learning: Naïve Bayes Algorithm. **Reinforcement Learning**: Overview and Scope of Reinforcement Learning, Components of Reinforcement Learning.

List of experiments

1. Develop a program to create histograms for all numerical features and analyze the distribution of each feature. Generate box plots for all numerical features and identify any outliers. Use California Housing dataset.
2. Develop a program to compute the correlation matrix to understand the relationships between pairs of features. Visualize the correlation matrix using a heatmap to know which variables have strong positive/negative correlations. Create a pair plot to visualize pairwise relationships between features. Use California Housing dataset.
3. Develop a program to implement Principal Component Analysis (PCA) for reducing the dimensionality of the Iris dataset from 4 features to 2.
4. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Find-S algorithm to output a description of the set of all hypotheses consistent with the training examples.
5. Develop a program to implement k-Nearest Neighbour algorithm to classify the randomly generated 100 values of x in the range of [0,1]. Perform the following based on dataset generated. a. Label the first 50 points {x1,……,x50} as follows: if (xi ≤ 0.5), then xi ∈ Class1, else xi ∈ Class1 b. Classify the remaining points, x51,……,x100 using KNN. Perform this for k=1,2,3,4,5,20,30
6. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.
7. Develop a program to demonstrate the working of Linear Regression and Polynomial Regression. Use Boston Housing Dataset for Linear Regression and Auto MPG Dataset (for vehicle fuel efficiency prediction) for Polynomial Regression.
8. Develop a program to demonstrate the working of the decision tree algorithm. Use Breast Cancer Data set for building the decision tree and apply this knowledge to classify a new sample.
9. Develop a program to implement the Naive Bayesian classifier considering Olivetti Face Data set for training. Compute the accuracy of the classifier, considering a few test data sets.
10. Develop a program to implement k-means clustering using Wisconsin Breast Cancer data set and visualize the clustering result.

Reference Books:

1. S. Sridhar and M. Vijayalakshmi, “Machine Learning”, Oxford University Press, 2021.
2. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Education, 2013.
3. Miroslav Kubat, “An Introduction to Machine Learning”, Springer, 2017.

Web links and Video Lectures (e-Resources):

- https://www.drssridhar.com/?page_id=1053
- https://www.universitiespress.com/resources?id=9789393330697
- https://onlinecourses.nptel.ac.in/noc23_cs18/preview
- https://www.geeksforgeeks.org/machine-learning/
- https://www.w3schools.com/python/python_ml_getting_started.asp
- https://www.tutorialspoint.com/machine_learning/index.htm

CO-PO Mapping:

Course Outcomes	Program Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	2 (WK1, WK2, WK3, WK4)	3 (WK, WK2, WK3, WK4)	3	3	--	--	--	--	--	--	1	2	1	3
CO2	3	2 (WK1, WK2, WK3, WK4)	3 (WK5)	2	2 (WK6)	--	--	--	1	1	--	1 (WK8)	2	3
CO3	--	2 (WK1, WK2, WK3, WK4)	3 (WK5)	--	3(WK6)	1 (WK1, WK5, WK7)	--	1	1	--	1 (WK8)	--	2	3
CO4	--	2 (WK1, WK2, WK3, WK4)	2 (WK5)	-	--	1 (WK1, WK5, WK7)	--	1	1	--	1 (WK8)	--	2	

Course Code		**:**	**1BUEA602C**	**CMOS VLSI Design**	**Semester**	**:**	**06**
L:T:P		**:**	**4:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**56 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**58**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120 Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Demonstrate the ability to analyze, design, and optimize CMOS VLSI circuits and array subsystems by applying principles of MOSFET physics, fabrication processes, and interconnect modeling to improve performance metrics such as delay and power dissipation.

Course Outcomes:

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

CO1: Draw the layout of CMOS circuits; apply the knowledge of fabrication processes and MOSFET transistors in VLSI design..

CO2: Draw an RC equivalent circuit of CMOS circuits and estimate delay and power.

CO3: Model & design of interconnects in chips, design of combinational circuits.

CO4: Design basic building blocks of sequential circuits and memory subsystems using MOSFET transistors.

Unit-I	**14 Hrs.**

Introduction: A Brief History, Preview, MOS Transistors, CMOS Logic, CMOS Fabrication and Layout, Design Partitioning. **MOS Transistor Theory**: Introduction, Long- Channel I-V Characteristics, C-V Characteristics (simple MOS capacitance models), Non ideal I-V Effects, DC Transfer Characteristics. **CMOS Processing Technology**: Introduction, CMOS Technologies.

Unit-II	**14 Hrs.**

Delay: Introduction, Transient Response, RC Delay Model, Linear Delay Model (Logical effort, parasitic delay, delay in logic gate, drive), Logical Effort of Paths, **Power**: Introduction, Dynamic Power, Static Power, Energy-Delay Tradeoffs, Power Reduction Techniques, Short-Circuit Power.

Unit-III	**14 Hrs.**

Interconnect: Introduction (wire Geometry), Interconnect Modeling, Interconnect Impact (Delay, Energy, Cross-talk). **Combinational Circuit Design**: Introduction, Circuit families, Silicon-On-Insulator Circuit Design.

Unit-IV	**14 Hrs.**

Sequential Circuit Design: Introduction, Circuit Design of Latches and Flip Flops (conventional CMOS latches, conventional CMOS flip flops, pulsed latches, resettable latches and flip flops, enabled latches and flip flops, incorporating logic into latches, dual edge triggered flip flops. **Array Subsystems**: Introduction, SRAM (SRAM cells, ROW circuitry, column circuitry), Read-Only Memory, Serial Access Memories, Content Addressable Memory, Programmable Logic Arrays.

Reference Books:

1. Neil H. E. Weste, David Harris, “CMOS VLSI Design A Circuits and Systems Perspective”, Pearson Education Publisher, 4th Edition, 2015.

2. Jan M. Rabaey, Anantha Chandrakasan, Borivoje Nikolic, “Digital Integrated Circuits A Design Perspective”, Pearson Education Publisher, 2nd Edition. 2010.
3. John P Uyemura, “Introduction to VLSI Circuits and Systems”, Wiley Publication 2002.
4. R. Jacob Baker, Harry W. Li, David E. Boyce, “CMOS Circuit Design, Layout, and Simulation”, Wiley.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	–	2 (WK2, WK6)	–	–	–	–	–	–	3	2	2
CO2	3 (WK3, WK4)	3 (WK3, WK4)	2 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	2	2
CO3	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	3
CO4	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3

Course Code		**:**	**1BUEA603C**	**Data Base Management System**	**Semester**	**:**	**06**
L:T:P		**:**	**3:0:2**		**Course Type**	**:**	**IPCC**
Hours/Sem.	**Teaching**	**:**	**42+26=68 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**44 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**120 Hrs.**		**Credits**	**:**	**04**

Professional Competency:

Apply database administration practices for backup, recovery, access control, and data security in enterprise environments.

Course Outcomes:

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

CO1: Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS.

CO2: Use of Structured Query Language (SQL) for Database Manipulation.

CO3: Design and build simple database systems.

CO4: Develop application to interact with databases.

Unit-I	**11 Hrs.**

Introduction to Databases: Introduction, Characteristics of Database Approach, Advantages of using the DBMS Approach, History of Database Applications, **Overview of Database Languages and Architectures:** Data Models, Schemas and Instances. Three schema Architecture and Data Independence, Database Languages and Interfaces, the Database System Environment, **Conceptual Data Modelling using Entities and Relationships:** Entity Types, Entity Sets, Attributes, Roles and Structural Constraints, Weak Entity Types, ER Diagrams, Examples, Specialization and Generalization, **Relational Model:** Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions and Dealing with Constraints Violations.

Unit-II	**10 Hrs.**

Relational Algebra: Unary and Binary Relational Operations, Additional Relational Operations, (aggregate, grouping, etc.) Examples of Queries in Relational Algebra, **SQL:** Advances Queries: More complex SQL Retrieval Queries, Specifying Constraints as Assertions and Action Triggers, Views in SQL, Schema change statements in SQL. **Database Application Development:** Accessing Databases from Applications, An Introduction to JDBC, JDBC Classes and Interfaces, SQLI, Stored Procedures and Case Study: The Internet Bookshop.

Unit-III	**11 Hrs.**

Internet Applications: The Three-Tier application Architecture, The Presentation Layer, The Middle Tier, **Normalization:** Database Design Theory – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for Relation Schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce- Codd Normal Form, Multivalued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, **Normalization Algorithms:** Inference Rules, Equivalence and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Nulls, Dangling, Tuples, Relational

Designs, Further discussion of Multivalued Dependencies and 4NF, Other Dependencies and Normal Forms.

Unit-IV	10 Hrs.

Transaction Processing: Introduction to Transaction Processing, Transaction and System Concepts, Desirable Properties of Transactions, Characterizing schedules based on Recoverability, Characterizing schedules based on Serializability, Transaction support in SQL, **Concurrency Control in Database:** Two-phase locking techniques for Concurrency Control, Concurrency Control based on Timestamp Ordering, Multiversion Concurrency Control Techniques, Validation Concurrency Control Techniques, Granularity of Data Items and Multiple Granularity Locking. **Introduction to Database Recovery Protocols:** Recovery Concepts, NO-UNDO/ REDO recovery based on Deferred Update, Recovery techniques based on Immediate Update, Shadow Paging, Database Backup and Recovery from Catastrophic Failures.

List of Experiments

1. Create a table called Employee & execute the following.

Employee (EMPNO, ENAME, JOB, MANAGER_NO, SAL, COMMISSION)

a) Create a user and grant all permissions to the user.
b) Insert the any three records in the employee table contains attributes EMPNO, ENAME JOB, MANAGER_NO, SAL, COMMISSION and use rollback. Check the result.
c) Add primary key constraint and not null constraint to the employee table.
d) Insert null values to the employee table and verify the result.

2. Create a table called Employee that contains attributes EMPNO, ENAME, JOB, MGR, SAL & execute the following.

a) Add a column commission with domain to the Employee table.
b) Insert any five records into the table.
c) Update the column details of job.
d) Rename the column of Employ table using alter command. 5. Delete the employee whose Empno is 105.

3. Queries using aggregate functions (COUNT, AVG, MIN, MAX, SUM) Group by, Order by. Employee (E_id, E_name, Age, Salary)

a) Create Employee table containing all Records E_id, E_name, Age, Salary.
b) Count number of employee names from employeetable.
c) Find the Maximum age from employee table.
d) Find the Minimum age from employee table.
e) Find salaries of employee in Ascending Order.
f) Find grouped salaries of employees.

4. Create a row level trigger for the customers table that would fire for INSERT or UPDATE or DELETE operations performed on the CUSTOMERS table. This trigger will display the salary difference between the old & new Salary.

CUSTOMERS (ID, NAME, AGE, ADDRESS, SALARY)

5. Create cursor for Employee table & extract the values from the table. Declare the variables, Open the cursor & extract the values from the cursor. Close the cursor.

Employee(E_id, E_name, Age, Salary)

6. Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table

already exist in the second table then that data should be skipped.

7. Install an Open Source NoSQL Data base MangoDB & perform basic CRUD (Create, Read, Update & Delete) operations. Execute MangoDB basic Queries using CRUD operations.

Reference Books:

1. Ramez Elmasri, Shamkant B. Navathe, " Fundamentals of Database Systems", 7th Edition, 2017, Pearson.
2. Ramakrishnan, Gehrke, "Database Manangement Systems", 3rd Edition, 2014, McGrawHill.
3. SilberschatzKorth, Sudharshan, "Database System Concepts", 6th edition, 2013, McGraw Hill.
4. Coronel, Morris, Rob, "Database Principles Fundamentals of Design, Implementation and Management", 2012, Cengage Learning

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	3 (WK3, WK4)	2 (WK5)	1 (WK8)	2 (WK2, WK6)	–	–	–	–	–	–	3	2	–
CO3	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	1
CO4	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	2

Course Code		**:**	**1BUEA604E**	**UI/UX Design**	**Semester**	**:**	**06**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**28 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design responsive and visually appealing interfaces by applying principles of typography, color theory, layout, navigation, and visual hierarchy

Course Outcomes:

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

CO1: Build UI for user applications.

CO2: Evaluate UX design of any product or application.

CO3: Demonstrate UX skills in product development.

CO4: Implement sketching principles.

Unit-I **07 Hrs.**

Foundations of Design: UI vs. UX Design – Core Stages of Design Thinking – Divergent and Convergent Thinking -Brainstorming and Game storming – Observational Empathy.

Unit-II **07 Hrs.**

Foundations of UI Design: Visual and UI Principles – UI Elements and Patterns – Interaction Behaviors and Principles -Branding – Style Guides.

Unit-III **07 Hrs.**

Foundation of UX Design: Introduction to User Experience - Why You Should Care About User Experience- Understanding User Experience- Defining The UX Design Process and its Methodology- Research in User Experience Design- Tools and Method Used For Research - User Needs and its Goals- Know About Business Goals.

Unit-IV **07 Hrs.**

Wireframing, Prototyping and Testing: Sketching Principles – Sketching Red Routes – Responsive Design – Wireframing – Creating Wireflows – Building a Prototype – Building High-Fidelity Mockups – Designing Efficiently with Tools – Interaction Patterns – Conducting Usability Tests – Other Evaluative User Research Methods -Synthesizing Test Findings – Prototype Iteration.

List of Experiments

1. Designing a Responsive layout for an societal application.
2. Exploring various UI Interaction Patterns.
3. Developing an interface with proper UI Style Guides.
4. Developing Wireflow diagram for application using open source software.
5. Exploring various open source collaborative interface Platform.
6. Hands on Design Thinking Process for a new product.
7. Brainstorming feature for proposed product.
8. Defining the Look and Feel of the new Project.
9. Create a Sample Pattern Library for that product (Mood board, Fonts, Colors based on UI principles).
10. Identify a customer problem to solve.
11. Conduct end-to-end user research – User research, creating personas, Ideation process (User stories, Scenarios), Flow diagrams, Flow Mapping.

12. Sketch, design with popular tool and build a prototype and perform usability testing and identify improvements.

13. Reference Books:

1. Joel Marsh, "UX for Beginners", 2022, O'Reilly.
2. Jon Yablonski, "Laws of UX using Psychology to Design Better Product and Services," 2021, O'Reilly.
3. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface", 3rd Edition, 2020, O'Reilly.
4. Steve Schoger, Adam Wathan, "Refactoring UI", 2018.
5. Steve Krug, "Don't Make Me Think, Revisited: A Commonsense Approach to Web & Mobile", 3rd Edition, 2015.

Web Links and Video Lectures (e-Resources):

1. https://www.nngroup.com/articles/
2. https://www.interaction-design.org/literature

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK1, WK2, WK3, WK4)	2 (WK1, WK2, WK3, WK4)	2 (WK3, WK5)	2 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	1	2
CO2	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	2 (WK3, WK5)	2 (WK3, WK4, WK8)	2 (WK2, WK6)	–	–	–	–	–	1 (WK8)	2	1	2
CO3	3 (WK1, WK2, WK3, WK4)	3 (WK1, WK2, WK3, WK4)	3 (WK3, WK5)	3 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	2	3
CO4	3 (WK1, WK2, WK3, WK4)	2 (WK1, WK2, WK3, WK4)	3 (WK3, WK5)	3 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	2	3

Course Code		**:**	**1BUEB604E**	**Full Stack Development**	**Semester**	**:**	**06**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**28+26= 54 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design and implement end-to-end web applications using contemporary full-stack technologies by integrating user interfaces, server-side programming, databases, APIs, authentication mechanisms, and deployment tools

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

CO1: Apply Javascript to build dynamic and interactive Web projects .

CO2: Implement user interface components for JavaScript-based Web using React.JS

CO3: Apply Express/Node to build web applications on the server side.

CO4: Develop data model in an open source nosql database and demonstrate modularization and packing of the front-end modules.

Unit-I	**10 Hrs.**

Introduction: Basic JavaScript Instructions, Statements, Comments, Variables, Data Types, Arrays, Strings, Functions, Methods & Objects, Decisions & Loops.**Document Object Model:** DOM Manipulation, Selecting Elements, Working with DOM Nodes, Updating Element Content & Attributes, Events, Different Types of Events, How to Bind an Event to an Element, Event Delegation, Event Listeners.

Unit-II	**11 Hrs.**

Form enhancement and validation**. Introduction to MERN:** MERN components, Server less Hello world. **React Components:** Issue Tracker, React Classes, Composing Components, Passing Data Using Properties, Passing Data Using Children, Dynamic Composition.

Unit-III	**10 Hrs.**

React State: Initial State, Async State Initialization, Updating State, Lifting State Up, Event Handling, Stateless Components, Designing Components, State vs. Props, Component Hierarchy, Communication, Stateless Components**. Express and GraphQL:** Express, REST API, GraphQL, Field Specification, Graph Based, Single Endpoint, Strongly Typed, Introspection, Libraries, The About API GraphQL Schema File, The List API, List APIIntegration, Custom Scalar types, The Create API, Create API Integration, Query Variables, Input Validations, Displaying Errors.

Unit-IV	**11 Hrs.**

MongoDB: Basics, Documents, Collections, Databases, Query Language, Installation, The Mongo Shell, **MongoDB CRUD Operations**: Create, Read, Projection, Update, Delete, Aggregate, **MongoDB Node.js Driver**: Schema Initialization, Reading from MongoDB, Writing to MongoDB. **Modularization and Webpack:** Back-End Modules Front-End Modules and Webpack Transform and Bundle, Libraries Bundle ,Hot Module Replacement, Debugging DefinePlugin: Build Configuration, Production Optimization.

List of Experiments

1. a. Write a script that Logs "Hello, World!" to the console. Create a script that calculates the sum of two numbers and displays the result in an alert box.
   b. Create an array of 5 cities and perform the following operations: Log the total number of cities. Add a new city at the end. Remove the first city. Find and log the index of a specific city.
2. a. Read a string from the user, Find its length. Extract the word "JavaScript" using substring() or slice(). Replace one word with another word and log the new string. Write a function isPalindrome(str) that checks if a given string is a palindrome (reads the same backward).
3. Create an object student with properties: name (string), grade (number), subjects (array), displayInfo() (method to log the student's details) Write a script to dynamically add a passed property to the student object, with a value of true or false based on their grade. Create a loop to log all keys and values of the student object.
4. Create a button in your HTML with the text "Click Me". Add an event listener to log "Button clicked!" to the console when the button is clicked. Select an image and add a mouseover event listener to change its border color. Add an event listener to the document that logs the key pressed by the user.
5. Build a React application to track issues. Display a list of issues (use static data). Each issue should have a title, description, and status (e.g., Open/Closed). Render the list using a functional component.
6. Create a component Counter with A state variable count initialized to 0. Create Buttons to increment and decrement the count. Simulate fetching initial data for the Counter component using useEffect (functional component) or componentDidMount (class component). Extend the Counter component to Double the count value when a button is clicked. Reset the count to 0 using another button.
7. Install Express (npm install express). Set up a basic server that responds with "Hello, Express!" at the root endpoint (GET /). Create a REST API. Implement endpoints for a Product resource: GET : Returns a list of products. POST : Adds a new product. GET /:id: Returns details of a specific product. PUT /:id: Updates an existing product. DELETE /:id: Deletes a product. Add middleware to log requests to the console. Use express.json() to parse incoming JSON payloads.
8. 8. Install the MongoDB driver for Node.js. Create a Node.js script to connect to the shop database. Implement insert, find, update, and delete operations using the Node.js MongoDB driver. Define a product schema using Mongoose. Insert data into the products collection using Mongoose. Create an Express API with a /products endpoint to fetch all products. Use fetch in React to call the /products endpoint and display the list of products. Add a POST /products endpoint in Express to insert a new product. Update the Product List, After adding a product, update the list of products displayed in React

Reference Books:

8. Jon Duckett, "JavaScript & jQuery: Interactive Front-End Web Development", Wiley, 2014.
9. Vasan Subramanian, Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node. Apress, 2019.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	3 (WK5)	1 (WK4, WK8)	3 (WK5, WK6)	1 (WK7)	–	1 (WK7)	1 (WK6)	2 (WK8)	–	3	2	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK4, WK8)	3 (WK5, WK6)	1 (WK7)	–	1 (WK7)	1 (WK6)	2 (WK8)	1 (WK6)	3	3	3
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK4, WK8)	3 (WK5, WK6)	1 (WK7)	–	2 (WK7)	2 (WK6)	2 (WK8)	1 (WK6, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK4, WK8)	3 (WK5, WK6)	2 (WK7)	–	2 (WK7)	2 (WK6)	2 (WK8)	2 (WK6, WK8)	3	3	3

Course Code		**:**	**1BUEC604E**	**Unix Shell Programming**	**Semester**	**:**	**06**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Proficiency in navigating the UNIX operating system, managing file systems, and developing automated solutions using shell scripting and advanced text-processing tools (AWK and Perl) to streamline software development and system administration tasks.

Course Outcomes:

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

CO1 : Understand the core UNIX architecture, navigate the UNIX file system, manage basic file attributes, and utilize the vi editor for text manipulation.

CO2 : Analyze shell behavior and process management, customize the user environment, and implement simple text filters along with extended file attributes.

CO3 : Apply regular expressions for advanced text filtering and construct essential shell scripts to automate system tasks.

CO4 : Develop advanced text processing and data manipulation scripts using the awk and perl programming tools.

Unit-I	**10 Hrs.**

The Unix Operating System, The UNIX Architecture and Command Usage, The File System, Basic File Attributes, The vi Editor.

Unit-II	**11 Hrs.**

The Shell, The Process, Customizing the Environment, More File Attributes, Simple Filters.

Unit-III	**10 Hrs.**

Filters using Regular Expressions, Essential Shell Programming.

Unit-IV	**11 Hrs.**

awk – An Advanced Filter (awk Programming for Pattern Scanning and Data Processing)

perl - The Master Manipulator (Perl Programming for Text Processing and Automation)

Reference Books:

1. Sumitabha Das: "UNIX – Concepts and Applications", 4th Edition, 2006, Tata McGraw Hill.
2. Behrouz A. Forouzan, Richard F. Gilberg, "UNIX and Shell Programming", 2005, Cengage Learning.
3. M.G. Venkateshmurthy, "UNIX & Shell Programming", 2005, Pearson Education.

CO-PO Mapping:

Course Outcomes:	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	1 (WK3, WK4)	–	–	2 (WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	2 (WK3, WK4)	–	–	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	–
CO3	2 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	1 (WK8)	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	2
CO4	2 (WK3, WK4)	2 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3

Course Code		**:**	**1BUED604E**	**Cyber Security**	**Semester**	**:**	**06**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Understand the fundamental principles of cyber security, information security, and risk management to protect digital assets.

Course Outcomes:

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

CO1: Describe the cybercrime terminologies.

CO2: Analyze cybercrime in mobiles and wireless devices along with the tools for cyber- crime and protection.

CO3: Analyze the motive and causes for cybercrime, cyber criminals and investigators.

CO4: Apply the methods for understanding criminal case and evidence, detection standing criminal case and evidence.

Unit-I	**10 Hrs.**

Introduction to Cybercrime: Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000. **Cyber offenses:** How Criminals Plan Them: Introduction, How Criminals Plan the Attacks.

Unit-II	**11 Hrs.**

Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes. Bot nets: The Fuel for Cybercrime, Attack Vector. Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares.

Unit-III	**10 Hrs.**

Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, Attacks on Wireless Networks. **Understanding the people on the scene:** Introduction, understanding cyber criminals, understanding cyber victims, understanding cyber investigators.

Unit-IV	**11 Hrs.**

Understanding Cybercrime Prevention: Understanding Network Security Concepts, Understanding Basic Cryptography Concepts, Making the Most of Hardware and Software Security. Cybercrime Detection Techniques: Security Auditing and Log: Auditing for Windows platform, Firewall Logs, Reports, Alarms, and Alerts, Commercial Intrusion Detection Systems.

Reference Books:

1. Sunit Belapure, Nina Godbole, “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, 2013, Wiley India Pvt Ltd, ISBN: 978-81- 265-21791.
2. Debra Little, John Shinder, Michael Cross, “Scene of the cybercrime”, 2nd Edition, 2008, Syngress publishing Inc, Elsevier Inc.

3. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1st Edition, 2016, O’Reilly Media, ISBN-13: 978-1491912058.
4. Charles Dierbach, “Introduction to Computer Science Using Python”, 1st Edition, 2015, Wiley India Pvt Ltd, ISBN-13: 978-8126556014.
5. Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, 2015, Pearson Education India, ISBN-13: 978-9332555365.

CO-PO Mapping:

Course Outcomes:	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	1 (WK1 WK2, WK3, WK4)	–	–	–	2 (WK2, WK6)	–	–	–	–	–	–	3	2	–
CO2	–	2 (WK3, WK4)	–	3 (WK3, WK4, WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	–
CO3	–	–	–	–	2 (WK2, WK6)	–	–	–	–	–	–	3	3	2
CO4	–	–	–	–	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3

Course Code		**:**	**1BUEA631N**	**Virtual Instrumentation**	**Semester**	**:**	**06**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**OEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Demonstrate the ability to design and implement virtual instrumentation systems using LabVIEW for data acquisition, monitoring, and control applications.

Course Outcomes:

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

CO1: Acquire the knowledge of virtual instrumentation, LabVIEW environment, and its applications.

CO2: Explain and develop modular LabVIEW programs using subVIs, loops, and related programming concepts.

CO3: Apply arrays and programming structures for developing LabVIEW applications.

CO4: Understand and implement data acquisition and motion control concepts using LabVIEW. understanding criminal case and evidence, detection standing criminal case and evidence.

Unit-I	**10 Hrs.**

Virtual Instrumentation: Virtual instrument and traditional instrument, hardware and software in VI, VI for test, control and design, VI in engineering process, Text based programming and graphical programming. **Introduction to LabVIEW**: Advantages, software environment, creating and saving VI, front panel and block diagram tool bar, palettes, controls and indicators, block diagram, data types.

Unit-II	**10 Hrs.**

Modular Programming: Build a VI front panel and block diagram, building a connector pane, displaying SubVIs and express VIs, creating SubVIs, **Repetition and Loops**: For loops, while loops, structure tunnels, terminal inside or outside loops, shift registers, feedback nodes, control timing, communication among multiple loops.

Unit-III	**11 Hrs.**

Arrays: Creating one dimensional, two dimensional, multi-dimensional arrays, array initialization, deleting, inserting, replacing elements within an array, array function, auto indexing. **Structures**: Case, sequence, sequence structure, formula nodes, event structures.

Unit-IV	**11 Hrs.**

Data Acquisition: Signals, signal conditioning, DAQ hardware configuration, DAQ hardware, analog inputs, outputs, counters. **Motion Control**: Components, software for configuration, prototyping and development, motion controller, move types, motor amplifiers and drives.

Reference Books:

1. Jerome, Jovitha, “Virtual instrumentation using LABVIEW”, 1st Edition, 2010, PHI.
2. Sanjay Gupta, Joseph John, “Virtual instrumentation using LABVIEW”, 2nd Edition, 2017, MGH.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	1 (WK3, WK4)	–	–	2 (WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	3	–
CO3	3 (WK3, WK4)	3 (WK3, WK4)	2 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	1
CO4	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	2

Course Code		**:**	**1BUEA605L**	**CMOS VLSI Laboratory**	**Semester**	**:**	**06**
L:T:P		**:**	**0:0:2**		**Course Type**	**:**	**PCCL**
Hours/Sem.	**Teaching**	**:**	**26 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**--**		**SEE Marks**	**:**	**50**
	Exam	**:**	**04 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**30 Hrs.**		**Credits**	**:**	**01**

Professional Competency:

Demonstrate the ability to design, simulate, verify, and implement CMOS/TG-based digital circuits using VLSI design tools by performing schematic capture, layout design, DRC/LVS verification, RC extraction, and performance analysis.

Course Outcomes:

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

CO1: Design CMOS/ TG based gates, MUX, flip flops, counters and shift register.

CO2: Draw layouts, perform DRC/ERC/LVS verification, and carry out DC, transient and post-layout analysis of CMOS circuits.

CO3: Design and simulate Transmission Gate (TG) based digital circuits for correct logical operation.

CO4: Design and verify sequential circuits such as shift registers and counters using EDA tools.

List of Experiments

Part - A: Full Custom IC Design Flow (Foundation)

For experiments 1 through 8, design the following CMOS circuits with the given specifications. Complete the VLSI design flow using the appropriate EDA tool:

a) Draw the schematic and verify the design using DC and Transient Analysis.
b) Draw the layout and perform DRC and ERC verification.
c) Perform LVS verification.
d) Perform RC extraction and back-annotation, and verify the design.

1. Inverter
2. Two-Input NAND Gate
3. Two-Input NOR Gate
4. Two-Input AND Gate
5. Two-Input OR Gate
6. Transmission Gate
7. Two-Input XOR Gate
8. Two-Input XNOR Gate

Part - B: Architectural Design and Simulation (Complex Systems)

For experiments 9 through 12, design the following circuits with the given specifications. Complete the front-end design flow using the appropriate EDA tool:

a) Draw the schematic and verify the design using DC and Transient Analysis.
Note: Layout and physical verification are not required for this section.

9. 2:1 Multiplexer (MUX)
10. 4:1 Multiplexer (MUX)
11. D- latch

12. D Flip-Flop
13. 3-Bit SISO Shift Register
14. 3-Bit Up Counter

Reference Books:

1. Neil H. E. Weste, David Harris "CMOS VLSI Design A Circuits and Systems Perspective", 4th Edition, 2015, Pearson Education Publisher.
2. Jan M. Rabaey, Anantha Chandrakasan, Borivoje Nikolic, "Digital Integrated Circuits- A Design Perspective" 2nd Edition, 2010, Pearson Education Publisher.
3. John P. Uyemura "Introduction to VLSI Circuits and Systems", 2002, Wiley Publication.
4. R. Jacob Baker, Harry W. Li, David E. Boyce, "CMOS Circuit Design, Layout, and Simulation", Wiley.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	2 (WK3, WK4)	3 (WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	2	3
CO2	3 (WK3, WK4)	3 (WK3, WK4)	2 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	–	–	–	–	3	3	3
CO3	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	3
CO4	3 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3

Course Code		:	1BHSA600C	Indian Knowledge System	Semester	:	06
L:T:P		:	1:0:0		Course Type	:	AEC
Hours/Sem.	Teaching	:	14Hrs		CIE Marks	:	50
	Learning (TW+SL)	:	13Hrs		SEE Marks	:	50
	Exam	:	03Hrs		Total Marks	:	100
	Total Hrs.	:	30Hrs		Credit	:	01

Professional Competency:

Apply the principles of the Indian Knowledge System to appreciate, preserve, and utilize traditional knowledge for addressing contemporary societal and professional challenges.

Course Outcomes:

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

CO1: Provide an overview of the concept of the Indian Knowledge System and its importance

CO2: Appreciate the need and importance of protecting traditional knowledge.

CO3: Recognize the relevance of Traditional knowledge in different domains.

CO4: Establish the significance of Indian Knowledge systems in the contemporary world

Unit-I	03 Hrs.

Indian Knowledge Systems (IKS)

Overview, Vedic Corpus, Philosophy in Indian Knowledge system.

Unit-II	04 Hrs.

Traditional Knowledge in Mathematics

Introduction to Indian Mathematics, Unique aspects of Indian Mathematics, Indian Mathematicians and their contribution. Number Systems and Units of Measurement.

Unit-III	04Hrs.

Traditional Knowledge in Science

Measurements for time, distance and weight, Astronomy, Indian contributions in astronomy, Astrology, The celestial coordinate system, Elements of the Indian calendar, Notion of years and month, Panchanga – The Indian calendar system

Unit-IV	04 Hrs.

Traditional Knowledge in Professional domain

Town Planning and Architecture, Agriculture, Governance and Public Administration

Text Books:

1. Mahadevan. B. Bhat Vinayak Rajat, Nagendra Pavana R.N. "Introduction to Indian Knowledge System: Concepts and Applications", PHI Learning Private Ltd. Delhi (2022**).** Pride of India: A Glimpse into India's Scientific Heritage, Samskrita Bharati, New Delhi

Reference Books:

1. Sampad and Vijay "The Wonder that is Sanskrit", Sri Aurobindo Society, Puducherry. (2011).
2. Acarya. P.K. Indian Architecture, MunshiramManoharlal Publishers, New Delhi. (1996).
3. Kapoor Kapil, Singh Avadhesh "Indian Knowledge Systems Vol – I & II", Indian Institute of Advanced Study, Shimla, H.P. (2021).
4. Dasgupta,S. A History of Indian Philosophy- Volume 1, Motilal Banarsidass, New Delhi. (1975).
5. PLofker. K. (1963). Mathematics in India, Princeton University Press, New Jeresy, USA"

Web Links and Video Lectures:
1. https://www.youtube.com/watch?v=LZP1StpYEPM 2. http://nptel.ac.in/courses/121106003/ 3.http://www.iitkgp.ac.in/department/KS;jsessionid=C5042785F727F6EB46CBF432D7683B63 (Centre of Excellence for Indian Knowledge System, IIT Kharagpur) 4. https://www.wipo.int/pressroom/en/briefs/tk_ip.html 5. https://unctad.org/system/files/official-document/ditcted10_en.pdf 6. http://nbaindia.org/uploaded/docs/traditionalknowledge_190707.pdf development goals/?gclid=EAIaIQobChMInpJtb_p8gIVTeN3Ch2 7. https://unfoundation.org/what-we-do/issues/sustainable developmentgoals/?gclid=EAIaIQobChMInp-

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	2 (WK3, WK4)	2 (WK3, WK4)	–	1 (WK4)		2 (WK7)	1 (WK8)	3 (WK9)	–	2 (WK8)	2 (WK8)	3	2	3
CO2	2 (WK3, WK4)	2 (WK3, WK4)	1 (WK3, WK8)	–	1 (WK2, WK6)	3 (WK7)	2 (WK8)	3 (WK9)	–	1 (WK8)	2(WK8)	3	3	3
CO3	2 (WK3, WK4)	2 (WK3, WK4)	1 (WK3, WK8)	–	1 (WK2, WK6)	3 (WK7)	3 (WK8)	3 (WK9)	1 (WK8)	2 (WK8)	3 (WK8)	3	3	3
CO4	2 (WK3, WK4)	2 (WK3, WK4)	–	–	–	2 (WK7)	2 (WK8)	3 (WK9)	–	2 (WK8)	2 (WK8)	3	3	3

Seventh semester Syllabus

Course Code		**:**	**1BHSA711C**	**Research Methodology and IPR**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**HSMC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to conduct ethical, systematic research and protect innovations through effective application of Intellectual Property Rights (IPR) principles.

Course Outcomes:

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

CO1: Integrate research methodology in engineering sciences in relevant trades.

CO2: Exhibit reflective thinking in problem solving exercises.

CO3: Identify criteria to fit one's own intellectual work in particular form of IPRs.

CO4: Develop skill of making search using modern tools and techniques and become patent agent through patent agent exam.

Unit-I	**10 Hrs.**

Introduction to Engineering Research: Fundamentals of Research: Meaning, objectives, and motivation in engineering research. **Types of Engineering Research:** Basic, applied, and translational research; identifying and solving Worthwhile problems. **Research Ethics:** Ethics in engineering research and practice, types of research misconduct, and ethical issues in authorship...

Unit-II	**10 Hrs.**

Literature Review and Citations: Technical Reading & Analysis: Methods for reviewing literature, analyzing prior art, and synthesizing new and existing knowledge. **Bibliographic Databases:** Web of Science, Google, Google Scholar, effective search strategies. **Conceptualizing Research:** Critical and creative reading, taking notes, reading mathematical models, algorithms, and datasheets. **Citations & Acknowledgments:** Attribution, citation styles, impact of keywords, citing datasets, and knowledge dissemination.

Unit-III	**10 Hrs.**

Fundamentals of Intellectual Property Rights and Patents: Introduction to Intellectual Property Rights (IPR): Concept, nature, and significance of Intellectual Property, **Types of Intellectual Property:** Patents, Copyrights, Trademarks, Industrial Designs, Geographical Indications, Trade Secrets, and Plant Varieties. **Role of IPR in innovation**: research, entrepreneurship, and economic development. **International IPR Framework:** Overview of WIPO, WTO, TRIPS Agreement, Paris Convention, and Berne Convention. **Patents:** Meaning, objectives, and characteristics of patents. Patentable and non-patentable inventions. **Criteria for patentability**: Novelty, Inventive Step, and Industrial Applicability. **Patent search**: Importance and databases (InPASS, PATENTSCOPE, Google Patents). **Patent filing procedure in India**: Provisional and Complete Specifications. Patent examination, publication, opposition, grant, renewal, and infringement.

Unit-IV	10 Hrs.

Copyrights, Trademarks, Industrial Designs and IPR Management: Copyrights: Meaning, subject matter, ownership, duration, and infringement,Copyright issues in the digital environment and academic research. **Trademarks and Industrial Designs:**Concept and importance of trademarks. Registration process, infringement, and remedies.Industrial Designs: Registration, protection, and significance. **Geographical Indications and Trade Secrets:**Concept, registration, and protection of Geographical Indications. Trade secrets and confidential information. **IPR Management and Commercialization:**Technology Transfer and Licensing. Patent valuation and commercialization. Intellectual Property in Start-ups and Academic Institutions. Case studies on IPR disputes and emerging challenges in Artificial Intelligence and Digital Technologies.

Case Studies on Patents. Case study of Curcuma (Turmeric) Patent, Case study of Neem Patent, CasestudyofBasmatipatent.**IPOrganizationsInIndia.SchemesandProgrammes**.

Reference Books:

1. Dr.Nejakar SantoshM,Dr.Bendigeri Harish (2023-24) "ResearchMethodology and Intellectual Property Rights",ISBN978-93-5987-928-4,
2. Thiel DavidV.,"ResearchMethodsforEngineers",CambridgeUniversityPress,
3. Acharya N.K., "IntellectualPropertyRights", 6thEdition, AsiaLawHouse.
4. P. Naryan, "Intellectual Property Law", 3rd Edition, 2007, Eastern Law House.
5. Dr. Myneni S. R., "Law of Intellectual Property", 9th Edition, 2019, Asia law House.
6. Dr. Reddy G.B, "Intellectual Property Rights and Law", Reprint Edition, 2020, Gogia Law Agency, Hydrabad.
7. N.R. Subbaram, S. Viswanathan, "Hand book Indian Patent Law and, Practice", 2008, Printers and Publishers Pvt. Ltd.
8. Cornish, "Intellectual Property Rights", Universal Publications.

Web Links and Video Lectures (e-Resources):

1.https://mooc.utm.my/courses/researchmethodologyforen?utm_source=chatgpt.com
2.https://arxiv.org/abs/physics/0601009?utm_source=chatgpt.com
3.https://scholar.google.com/?utm_source=chatgpt.com
4.https://www.webofscience.com/?utm_source=chatgpt.com
5.https://www.wipo.int/en/web/wipo-academy/programs/ip-elearning?utm_source=chatgpt.com
6.https://ipindia.gov.in/?utm_source=chatgpt.com
7.https://patentscope.wipo.int/?utm_source=chatgpt.com
8.https://patents.google.com/?utm_source=chatgpt.com
9.https://nptel.ac.in/courses/109106148?utm_source=chatgpt.com

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	–	2 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	1 (WK8)	2 (WK7)	3 (WK8)	–	1 (WK8)	2 (WK8)	1	–	–
CO2	–	3 (WK3, WK4)	3 (WK5)	–	3 (WK2, WK6)	2 (WK8)	2 (WK7)	2 (WK8)	–	–	2 (WK8)	2	–	–
CO3	–	–	–	2 (WK8)	3 (WK2, WK6)	3 (WK8)	2 (WK7)	3 (WK8)	–	1 (WK8)	3 (WK8)	1	–	–
CO4	–	–	2 (WK5)	3 (WK8)	3 (WK2, WK6)	2 (WK8)	–	1 (WK8)	–	1 (WK8)	–	2	–	–

Course Code		**:**	**1BUEA702C**	**Big Data Analysis**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PCC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to analyze, process, and visualize large-scale datasets using modern big data technologies to support data-driven decision-making and intelligent business solutions.

Course Outcomes:

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

CO1: Understand the fundamentals of big data analysis and Hadoop.

CO2: Demonstrate the use of Hadoop and NoSQL.

CO3: Use MangoDB, Cassandra and MapReduce technologies for big data analysis

CO4: Exhibit the ability to use algorithms for data analysis and text, web content, link, and Social network analytics.

Unit-I	**10 Hrs.**

Introduction to Big Data Analytics: Big data, Scalability and parallel processing, Designing data architecture, Data sources, Quality, Pre-processing and storing, Data storage and analysis, Big data analytics Applications and case studies. **Introduction to Hadoop:** Introduction, Hadoop and its ecosystem, Hadoop distributed file system, MapReduce framework and programming model, Hadoop yarn, Hadoop ecosystem tools.

Unit-II	**11 Hrs.**

Hadoop Distributed File System Basics: HDFS design features, Components, HDFS user commands. **Essential Hadoop Tools:** Using Apache Pig, Hive, Sqoop, Flume, Oozie, HBase. **NoSQL Big Data Management:** Introduction, NoSQL data store, NoSQL data architecture patterns, NoSQL to manage big data, Shared-Nothing architecture for big data tasks.

Unit-III	**11 Hrs.**

MongoDB and Cassandra: MongoDB, Databases, Cassandra Databases. **Introduction to MapReduce:** Introduction, MapReduce map tasks, Reduce tasks and MapReduce execution, Composing MapReduce for calculations and algorithms, Hive, HiveQL, Pig. **Machine Learning Algorithms for Big Data Analytics:** Introduction, Estimating the relationships, Outliers, Variances, Probability distributions, and correlations.

Unit-IV	**10 Hrs.**

Algorithms for Big Data Analytics: Regression analysis, Finding similar items, Similarity of sets and collaborative filtering, Frequent item sets and association rule mining. **Text, Web Content, Link, and Social Network Analytics:** Introduction, Text mining, Web mining, Web content and web usage Analytics, Page rank, Structure of web and analyzing a web graph, Social network as graphs and Social network analytics.

Reference Books:

1. Raj Kamal and Preeti Saxena, “Big Data Analytics: Hadoop, Spark, and Machine Learning,” 2018, McGraw Hill Education,ISBN: 9789353164966, 9353164966.
2. Douglas Eadline, “Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem,” 1st Edition, 2016, Pearson Education, ISBN-13: 978-9332570351.
3. Tom White, “Hadoop: The Definitive Guide,” 4th Edition, 2015, O’Reilly Media, ISBN-13: 978-9352130672.
4. Boris Lublinsky, Kevin T Smith, Alexey Yakubovich, "Professional Hadoop Solutions," 1st Edition, 2014, Wrox Press, ISBN-13: 978-8126551071.
5. Eric Sammer, "Hadoop Operations: A Guide for Developers and Administrators," 1st Edition, 2012, O'Reilly Media, ISBN-13: 978-9350239261.
6. Arshdeep Bahga, Vijay Madisetti, "Big Data Analytics: A Hands-On Approach", 1st Edition, 2018, VPT Publications, ISBN-13: 978-0996025577.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK 3, WK 4)	2 (WK3 , WK4)	1 (WK 5)	2 (WK 8)	–	–	–	–	–	–	1 (WK 8)	3	1	2
CO2	3 (WK 3, WK 4)	3 (WK3 , WK4)	2 (WK 5)	2 (WK 8)	–	–	–	–	–	–	1 (WK 8)	2	1	2
CO3	3 (WK 3, WK 4)	3 (WK3 , WK4)	3 (WK 5)	3 (WK 8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK 3, WK 4)	2 (WK3 , WK4)	2 (WK 5)	3 (WK 8)	–	–	–	–	–	–	2 (WK 8)	3	2	3

Course Code		**:**	**1BUEA703C**	**Computer Networks**	**Semester**	**:**	**07**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**IPCC**
Hours/Sem.	**Teaching**	**:**	**28+26=54 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, analyze, configure, and evaluate computer networks by applying networking principles, communication protocols, and modern networking tools to develop reliable, secure, and efficient communication systems.

Course Outcomes:

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

CO1: Explain the terminology and concepts of the OSI reference model and the TCP/IP reference model.

CO2: Master the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks.

CO3: Identify, compare and contrast different techniques and design issues of core functions such as addressing, routing, internetworking, switching, multiplexing, error and flow control, medium access and coding.

CO4: Familiarize with widely- used Internet protocols such as TCP/IP, UDP and DNS in the Internet.

Unit-I	**07 Hrs.**

Layered tasks, OSI Model, Layers in OSI model, TCP/IP Suite, Addressing, Data Link Control: Framing, Flow and error control, Protocols, Noiseless channels and noisy channels, HDLC, PPP.

Unit-II	**07 Hrs.**

Multiple Accesses: Random access, Controlled access, Channelization, Wired LAN, Ethernet, IEEE standards, Standard Ethernet. Changes in the standards, Fast Ethernet, Gigabit Ethernet, Connecting LANs, Backbone and Virtual LANs.

Unit-III	**07 Hrs.**

Network Layer, Logical addressing, Ipv4 addresses, Ipv6 addresses, Ipv4 and Ipv6 Transition from Ipv4 to Ipv6, Delivery, Forwarding, Unicast Routing Protocols, Multicast Routing protocols.

Unit-IV	**07 Hrs.**

Transport layer Process to process Delivery, UDP, TCP, Application Layer: Domain name system, Name Space, Domain Name Space, Distribution of Name Space, DNS in the Internet, Resolution, DNS messages, Types of Records, Registrars, Dynamic Domain Name System, Encapsulation.

List of Experiments

1. Connecting Computers on LAN.
2. Study of packet tracer.
3. Configuration of different network topologies using packet tracer.
4. Configuration of switches and establishing LAN using packet tracer.
5. Creation of Virtual LAN using packet tracer.

6. Configuration of basic routing using packet tracer.
7. Configuration of a network using Routing Information Protocal (RIP) using packet tracer.
8. Configuration of a network using Open Shortest Path First (OSPF) using packet tracer.
9. Configuration of DHCP using packet tracer.
10. Configuration of NAT using packet tracer.

Reference Books:

1. Behrouz A. Forouzan, "Data Communication and Networking", 4th Edition, 2006, TMH, India.
2. Andrew S. Tanenbaum, "Computer Networks", 2010, Prentice-Hall.
3. William Stallings, "Data and Computer Communications", 2007, Prentice-Hall.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	–	2 (WK2, WK6)	–	–	–	–	–	1 (WK4)	3	2	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	1 (WK8)	3 (WK2, WK6)	1 (WK7)	–	1	1	–	2 (WK4, WK6)	3	3	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	1 (WK5, WK7)	–	1	1	–	2 (WK4, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK5)	1 (WK8)	3 (WK2, WK6)	1 (WK7)	–	1	1	–	2 (WK4, WK6)	3	3	3

Course Code		**:**	**1BUEA704E**	**System Verilog**	**Semester**	**:**	**07**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**28+26=54 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, model, verify, and validate digital hardware systems using System Verilog and modern Electronic Design Automation (EDA) tools by applying hardware design and verification methodologies to develop reliable and efficient digital solutions

Course Outcomes:

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

CO1: Explain the principles and methodologies used in verifying the functionality of digital systems.

CO2: Apply the System Verilog concepts to verify the complex digital circuits and develop and implement constrained random test benches using System Verilog

CO3: Analyze verification quality using functional coverage metrics and implement cover groups

CO4: Analyze real-world verification scenarios and apply System Verilog methodologies

Unit-I	**07 Hrs.**

Introduction to System Verilog: System Verilog standards, Key System Verilog enhancements for hardware design. Advantages of System Verilog over Verilog. **Data Types:** Verilog data types, System Verilog data types, 2 - State Data types, Bit, byte, shortint, int, longint. 4 - State data types. Logic, Enumerated data types, User Defined data types, Struct data types, Strings, Packages, Type Conversion: Dynamic casting, Static Casting, Memories: Arrays, Dynamic Arrays, Multidimensional Arrays, Packed Arrays, Associative Arrays, Queues, Array Methods.

Unit-II	**07 Hrs.**

Connecting the Test Bench and Design: Separating the Test Bench and Design, The Interface Construct, Stimulus Timing, Interface Driving and Sampling, System Verilog Assertions. **Procedural Statements and Routines:** Procedural Statements, Tasks, Functions and Void Functions, Task and Function Overview, Routine Arguments, Returning from a Routine, Local Data Storage, Time Values.

Unit-III	**07 Hrs.**

Randomization: Introduction, Randomization in System Verilog, Constraint Details, Solution Probabilities, Valid Constraints, InLine Constraints, Random Number Functions, Common Randomization Problems, Random Control, Random Number Generators

Unit-IV	**07 Hrs.**

Functional Coverage: Introduction, Coverage Types, Functional Coverage Strategies , Anatomy of a Cover Group, Triggering a Cover Group, Data Sampling, Cross Coverage, Coverage Options, Parameterized Cover Groups. **Complete Design Model Using System Verilog- Case Study**: System Verilog ATM Example, Data Abstraction, Interface Encapsulation, Design Top Level Squat, Receivers and Transmitters, Test Bench for ATM.

List of Experiments

1. Develop mux2x1 verilog code in design.sv and develop Test bench for mux2x1 in testbench.sv with different inputs at various times.
2. Write a program to demonstrate two-state and four-state data types.
3. Write a program to demonstrate push_front, pop_front, push_back and pop_back with respect to Queues.
4. Design a full adder using the interface construct to model inter-module communication.
5. Create a program to highlight the difference between rand and randc constructs in constrained randomization.
6. Build a 4-bit adder and verify its functionality using a structured verification environment.
7. Design a UART module using SystemVerilog: a. Develop the transmitter logic b. Implement the receiver logic c. Create a top-level testbench to verify UART operation
8. Design, simulate, and analyze functional coverage to ensure thorough verification of a given design.
9. Design and simulate a verification test that exercises multiple sequences.
10. Create and simulate a configurable Universal Verification Methodology (UVM) test environment.

Reference Books:

1. Chris Spear, "System Verilog for Verification – A guide to learning the Test bench language features", Springer Publications, 2nd Edition, 2010.
2. Stuart Sutherland, Simon Davidmann, Peter Flake, "System Verilog for Design- A guide to using System Verilog for Hardware Design and Modelling", 2nd Edition, 2006, Springer Publications,
3. Janick Bergeron, "Writing Test benches using System Verilog".
4. Doulos, "System Verilog Golden Reference Guide-A Concise Guide to System Verilog", IEEE Standard1800- 2009, Version 5.0, ISBN: 0-9547345-9-9, 2012.
5. SasanIman, Hansen Brown "Step-by-Step Functional Verification with System Verilog and OVM", 2008, Publishing Company, ISBN-13: 978-0-9816-5621-2.
6. Janick Bergeron, "Verification Methodology Manual".

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	1 (WK8)	2 (WK2, WK6)	–	–	–	–	–	1 (WK4)	3	2	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	3 (WK2, WK6)	–	–	1	1	–	2 (WK6, WK8)	3	3	3
CO4	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	3 (WK2, WK6)	1 (WK5, WK7)	–	1	1	–	2 (WK4, WK6, WK8)	3	3	3

Course Code		**:**	**1BUEB704E**	**Block Chain Technology**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to design, implement, and manage secure, decentralized blockchain-based applications using distributed ledger technologies and smart contracts.

Course Outcomes:

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

CO1: Explain the blockchain terminologies with its applications.

CO2: Illustrate the working principles of blockchain and the smart contract lifecycle.

CO3: Demonstrate the principles and methodologies used in Bitcoin.

CO4: Explain various blockchain platforms and tools and make use of Hyperledger in blockchain application architecture.

Unit-I	**10 Hrs.**

Distributed systems, CAP theorem, Byzantine Generals problem, Consensus. The history of blockchain, Introduction to blockchain, Various technical definitions of blockchains, Generic elements of a blockchain, Features of a blockchain, Applications of blockchain technology, Tiers of blockchain technology, Consensus in blockchain, CAP theorem and blockchain, Benefits and limitations of blockchain.

Unit-II	**11 Hrs.**

Decentralization using blockchain, Methods of decentralization, Blockchain and full ecosystem decentralization, Smart contract, Decentralized organizations, Decentralized autonomous organizations, Decentralized autonomous corporations, Decentralized autonomous societies Decentralized applications, Platforms for decentralization. Cryptographic primitives: Symmetric cryptography, Asymmetric cryptography, Public and private keys, Hash functions: Compression of arbitrary messages into fixed length digest, Easy to compute, Pre-image resistance, Second pre-image resistance, Collision resistance, Message Digest (MD),Secure Hash Algorithms (SHAs), Merkle trees, Patricia trees, Distributed hash tables (DHTs), Digital signatures, Elliptic Curve Digital signature algorithm (ECDSA).

Unit-III	**11 Hrs.**

Bitcoin, Bitcoin definition, Transactions, The transaction life cycle, The transaction structure, Types of transaction, The structure of a block , The structure of a block header, The genesis block, The bitcoin network, Wallets, Smart Contracts-History, Definition, Ricardian contracts, Smart contract templates, Oracles, Smart Oracles, Deploying smart contracts on a blockchain, The DAO.

Unit-IV	**10 Hrs.**

Ethereum 101, Introduction, Ethereum clients and releases, The Ethereum stack, Ethereum blockchain, Currency (ETH and ETC), Forks, Gas, The consensus mechanism, The world state, Transactions, Contract creation transaction, Message call transaction, Elements of the Ethereum blockchain , Ethereum virtual

machine (EVM), Accounts, Block, Ether, Messages, Mining, The Ethereum network. Hands-on: Clients and wallets –Geth. Hyperledger, Hyperledger as a protocol, Fabric, Hyperledger Fabric, Sawtooth lake, Corda.

Reference Books:

1. Imran Bashir, "Mastring Blockchain", 3rd Edition, 2020, Packt.
2. Andreas M., "Mastering Bitcoin: Programming the Open Blockchain", 2017, O'Rielly.

Co-Po Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2(WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		**:**	**1BUEC704E**	**Cloud Computing**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to design, deploy, and manage scalable, secure, and cost-effective cloud-based applications and services using modern cloud computing technologies.

Course Outcomes:

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

- **CO1:** Understand various platforms which use modern cloud computing systems.
- **CO2:** Demonstrate the concepts of virtualization technologies along with the architectural models of cloud computing.
- **CO3:** Describe the concept of cloud computing architecture and cloud security.
- **CO4:** Understand how industries leverage various cloud technologies.

Unit-I	**10 Hrs.**

Introduction: Introduction, Cloud computing at a glance, Historical developments, Building cloud computing environments, Amazon Web Services (AWS), Google App Engine, Microsoft Azure, Hadoop, Force.com, Salesforce.com, Manjrasoft Aneka.

Unit-II	**11 Hrs.**

Virtualization: Introduction, Characteristics of virtualized, Environments taxonomy of virtualization techniques, Execution virtualization, Other types of virtualization, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples.

Unit-III	**10 Hrs.**

Cloud Computing Architecture: Introduction, Cloud reference model, Types of clouds, Economics of the cloud, Open challenges. **Cloud Security:** Risks, Top concern for cloud users, privacy impact assessment, Trust, OS security, VM security, Security risks posed by shared images and management OS.

Unit-IV	**11 Hrs.**

Cloud Platforms in Industry: Amazon Web Services: Compute services, Storage services, Communication services, Additional services. **Google AppEngine:** Architecture and core concepts, Application life cycle, Cost model, Observations. **Cloud Applications:** Scientific applications, HealthCare - ECG analysis in the cloud, Biology - Gene expression data analysis for cancer diagnosis, Geo-science -Satellite image processing. Business and consumer applications - CRM and ERP, Social networking, Media applications.

Reference Books:

1. Rajkumar Buyya, Christian Vecchiola, Thamrai Selvi, “Mastering Cloud Computing”, Morgan Kaufmann.
2. Dan C. Marinescu, “Cloud Computing Theory and Practice”, Morgan Kaufmann, 2013, Elsevier.
3. Toby Velte, Anthony Velte, “Cloud Computing: A Practical Approach”, McGraw-Hill Osborne Media.

4. George Reese, “Cloud Application Architectures: Building Applications and Infrastructure in the Cloud”, O'Reilly Publication.
5. John Rhoton, “Cloud Computing Explained: Implementation Handbook for Enterprises”, Recursive Press.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2(WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		**:**	**1BUED704E**	**Software Engineering and Project Management**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to design, develop, test, and manage software projects by applying software engineering principles, project management practices, and industry standards to deliver high-quality software solutions.

Course Outcomes:

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

CO1: Understand the activities involved in software engineering and analyze the role of various process models.

CO2: Explain the basics of object-oriented concepts and build a suitable class model using modeling techniques.

CO3: Describe various software testing methods and to understand the importance of Agile methodology and DevOps and illustrate the role of project planning and quality management in software development.

CO4: Understand the importance of activity planning and different planning models.

Unit-I	**10 Hrs.**

Software and Software Engineering: The nature of software, The unique nature of WebApps, Software engineering, The software process, The software engineering practice, The software myths, How it all starts. **Process Models:** A generic process model, Process assessment and improvement, Prescriptive process models, Waterfall model, Incremental process models, Evolutionary process models, Concurrent models, Specialized process models.

Unit-II	**11 Hrs.**

Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling. **Process Synchronization:** Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors, **Deadlocks:** Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

Unit-III	**11 Hrs.**

Memory Management: Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation, **Virtual Memory Management:** Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System, Implementation of File System:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management.

Unit-IV	10 Hrs.
Secondary Storage Structures, Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems, **Case Study:** The Linux Operating System: Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication.	

Reference Books:

1. Roger S. Pressman, "Software Engineering-A Practitioners approach", 7th Edition, Tata McGraw Hill.
2. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", 6th Edition, 2018, McGraw Hill Education.
3. Pankaj Jalote, "An Integrated Approach to Software Engineering", Wiley India.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	2	3	4	5	6	7	8	9	10	11	1	2	3
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	2 (WK5)	–	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2(WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	3 (WK5)	2 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		**:**	**1BUEA705E**	**Data Analytics Using R**	**Semester**	**:**	**07**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**28+26=54 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to analyze, visualize, and interpret data using R programming to extract meaningful insights and support data-driven decision-making.

Course Outcomes:

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

CO1: Describe the structures of R Programming

CO2: Illustrate the basics of Data Preparation with real world examples

CO3: Apply the Graphical Packages of R for visualization

CO4: Apply various Statistical Analysis methods for data analytics.

Unit-I	**07 Hrs.**

Basics of R: Introducing R, Initiating R, Packages in R, Environments and Functions, Flow Controls, Loops, Basic Data Types in R, Vectors, Matrices and Arrays, Lists

Unit-II	**07 Hrs.**

Basics of R (Continued): Data Frames, Factors, Strings, Dates and Times, **Data Preparation:** Datasets, Importing and Exporting files, Accessing Databases, Data Cleaning and Transformation

Unit-III	**07 Hrs.**

Graphics using R: Exploratory Data Analysis, Main Graphical Packages, Pie Charts, Scatter Plots, Line Plots, Histograms, Box Plots, Bar Plots, Other Graphical packages

Unit-IV	**07 Hrs.**

Statistical Analysis using R: Basic Statistical Measures, Normal distribution, Binomial distribution, Correlation Analysis, Regression Analysis-Linear Regression Analysis of Variance

List of Experiments

1. Demonstrate the steps for installation of R and R Studio. Perform the following:
   a) Assign different type of values to variables and display the type of variable. Assign different types such as Double, Integer, Logical, Complex and Character and understand the difference between each data type.
   b) Demonstrate Arithmetic and Logical Operations with simple examples.
   c) Demonstrate generation of sequences and creation of vectors.
   d) Demonstrate Creation of Matrices
   e) Demonstrate the Creation of Matrices from Vectors using Binding Function.

Demonstrate element extraction from vectors, matrices and arrays

2. Assess the Financial Statement of an Organization being supplied with 2 vectors of data: Monthly

Revenue and Monthly Expenses for the Financial Year. You can create your own sample data vector for this experiment) Calculate the following financial metrics:

a. Profit for each month.
b. Profit after tax for each month (Tax Rate is 30%).
c. Profit margin for each month equals to profit after tax divided by revenue.
d. Good Months – where the profit after tax was greater than the mean for the year.
e. Bad Months – where the profit after tax was less than the mean for the year.
f. The best month – where the profit after tax was max for the year.
g. The worst month – where the profit after tax was min for the year.

Note:

a. All Results need to be presented as vectors
b. Results for Dollar values need to be calculated with $0.01 precision, but need to be presented in Units of $1000 (i.e 1k) with no decimal points
c. Results for the profit margin ratio need to be presented in units of % with no decimal point.
d. It is okay for tax to be negative for any given month (deferred tax asset)

Generate CSV file for the data.

3. Develop a program to create two 3 X 3 matrices A and B and perform the following operations
a) Transpose of the matrix b) addition c) subtraction d) multiplication

4. Develop a program to create two 3 X 3 matrices A and B and perform the following operations a) Transpose of the matrix b) addition c) subtraction d) multiplication

5. Develop an R Program using functions to find all the prime numbers up to a specified number by the method of Sieve of Eratosthenes.

6. The built-in data set mammals contain data on body weight versus brain weight. Develop R commands to: a) Find the Pearson and Spearman correlation coefficients. Are they similar? b) Plot the data using the plot command. c) Plot the logarithm (log) of each variable and see if that makes a difference

7. Develop R program to create a Data Frame with following details and do the following operations.

Item code	Item category	Item price
1001	Electronics	700
1002	Desktop Supplies	300
1003	Office Supplies	350
1004	USB	400
1005	CD Drive	800

Subset the Data frame and display the details of only those items whose price is greater than or equal to 350.

Subset the Data frame and display only the items where the category is either “Office Supplies” or “Desktop Supplies”

Create another Data Frame called “item-details” with three different fields itemCode, ItemQtyonHand and ItemReorderLvl and merge the two frames

8. Let us use the built-in dataset air quality which has Daily air quality measurements in New York, May toSeptember 1973. Develop R program to generate histogram by using appropriate arguments for the following statements.

a) Assigning names, using the air quality data set.
b) Change colors of the Histogram

c) Remove Axis and Add labels to Histogram

d) Change Axis limits of a Histogram

e) Add Density curve to the histogram

9.Design a data frame in R for storing about 20 employee details. Create a CSV file named "input.csv" that defines all the required information about the employee such as id, name, salary, start_date, dept. Import into R and do the following analysis.

Find the total number rows & columns b) Find the maximum salary c) Retrieve the details of the employee with maximum salary d) Retrieve all the employees working in the IT Department. e) Retrieve the employees in the IT Department whose salary is greater than 20000 and write these details into another file "output.csv".

10. Using the built in dataset mtcars which is a popular dataset consisting of the design and fuel consumption patterns of 32 different automobiles. The data was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973-74 models). Format A data frame with 32 observations on 11 variables : [1] mpg Miles/(US) gallon, [2] cyl Number of cylinders [3] disp Displacement (cu.in.), [4] hp Gross horsepower [5] drat Rear axle ratio,[6] wt Weight (lb/1000) [7] qsec 1/4 mile time, [8] vs V/S, [9] am Transmission (0 = automatic, 1 = manual), [10] gear Number of forward gears, [11] carb Number of carburetors

Develop R program, to solve the following:

a) What is the total number of observations and variables in the dataset?

b) Find the car with the largest hp and the least hp using suitable functions

c) Plot histogram / density for each variable and determine whether continuous variables are normally distributed or not. If not, what is their skewness?

What is the average difference of gross horse power(hp) between automobiles with 3 and 4 number of cylinders(cyl)? Also determine the difference in their standard deviations. e) Which pair of variables has the highest Pearson correlation?

11. Demonstrate the progression of salary with years of experience using a suitable data set (You can create your own dataset). Plot the graph visualizing the best fit line on the plot of the given data points. Plot a curve of Actual Values vs. Predicted values to show their correlation and performance of the model. Interpret the meaning of the slope and y-intercept of the line with respect to the given data. Implement using lm function. Save the graphs and coefficients in files. Attach the predicted values of salaries as a new column to the original data set and save the data as a new CSV file.

Reference Books:

1. G. Sudhamathy, C. Jothi Venkateswaran, "R Programming: An Approach to Data Analytics", 2019, MJP Publishers
2. W. N. Venables, D.M. Smith, R development core team, "An Introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics", Version 3.0.1 (2013-05-16).
1. Cotton R.,"Learning R: A Step by Step Function Guide to Data Analysis", 1st Edition, O'Reilly Media Inc.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		**:**	**1BUEB705E**	**Robotics and Automation**	**Semester**	**:**	**07**
L:T:P		**:**	**2:0:2**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**28+26= 54 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**28 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**08 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Design, program, and troubleshoot robotic systems for industrial and service applications.

Course Outcomes:

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

CO1: Explain robot fundamentals, classification, anatomy and applications.

CO2: Apply coordinate transformations and kinematic analysis techniques to robotic manipulators.

CO3: Analyze robot actuators, sensors and dynamic behavior.

CO4: Describe robot control strategies, programming methods and emerging robotic technologies.

Unit-I	**10 Hrs.**

Fundamentals of Robotics: Introduction to Robotics: Definition, History and Evolution of Robots, Classification of Robots, Components of Robotic Systems, Robot Anatomy and Degrees of Freedom, Robot Configurations, Robot Workspace, Robot Applications in Manufacturing, Healthcare, Agriculture, and Service Industries, Robot Performance Parameters.

Unit-II	**11 Hrs.**

Robot Kinematics: Coordinate Systems and Transformations, Position and Orientation Representation, Homogeneous Transformation Matrices, Forward Kinematics, Denavit - Hartenberg (D-H) Representation, Inverse Kinematics Concept, Kinematic Analysis of Simple Manipulators, Jacobian Matrix and Velocity Analysis.

Unit-III	**10 Hrs.**

Robot Dynamics, Actuators and Sensors: Introduction to Robot Dynamics, Equations of Motion, Actuation Systems: Electric, Hydraulic and Pneumatic Drives, DC and Servo Motors, Stepper Motors, End Effectors and Grippers, Sensors in Robotics: Position, Velocity, Force/Torque, Proximity and Vision Sensors, Sensor Integration.

Unit-IV	**11 Hrs.**

Robot Control and Applications: Robot Control Systems, Open-loop and Closed-loop Control, PID Control Basics, Robot Programming Concepts, Programming Methods: Lead-through and Off-line Programming, Industrial Robot Applications, Mobile Robots, Collaborative Robots (Cobots), Autonomous Robots, Introduction to Artificial Intelligence in Robotics, Emerging Trends and Future Directions.

List of Experiments

1. Visualize the Denavit–Hartenberg (DH) parameters of a serial manipulator using RoboAnalyzer.
2. Perform forward kinematics of a robot manipulator for a given set of joint variables.
3. Determine inverse kinematics for a desired end-effector position and orientation.
4. Verify forward kinematics by comparing the computed end-effector pose with RoboAnalyzer

animation.

5. Generate and analyze a joint-space trajectory between two robot configurations.
6. Compute inverse dynamics to obtain the actuator torques for a specified robot motion.
7. Perform forward dynamics simulation of a manipulator using specified joint torques.
8. Simulate robot motion using the Virtual Robot module and observe the manipulator movement.
9. Study recursive inverse dynamics of a serial manipulator using the RoboAnalyzer module.
10. Study recursive forward dynamics and simulation using RoboAnalyzer and analyze the resulting joint

Reference Books:

1. S. K. Saha, "Introduction to Robotics", 3rd Edition, McGraw-Hill Education.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		**:**	**1BUEC705E**	**Cryptography and Network Security**	**Semester**	**:**	**07**
L:T:P		**:**	**3:0:0**		**Course Type**	**:**	**PEC**
Hours/Sem.	**Teaching**	**:**	**42 Hrs.**		**CIE Marks**	**:**	**50**
	Learning (TW+SL)	**:**	**42 Hrs.**		**SEE Marks**	**:**	**50**
	Exam	**:**	**06 Hrs.**		**Total Marks**	**:**	**100**
	Total Hrs.	**:**	**90 Hrs.**		**Credits**	**:**	**03**

Professional Competency:

Develop the ability to design and implement secure communication systems by applying cryptographic techniques and network security principles to protect data and information systems from cyber threats.

Course Outcomes:

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

CO1: Analyze different terminologies of cryptography.

CO2: Understand the block cipher principle and encryption standards.

CO3: Describe public key cryptosystem and key management techniques

CO4: Understand IP security architecture, summarize web security and system security concepts.

Unit-I	**10 Hrs.**

Introduction: OSI security architecture, Security attacks, Security services, Security mechanism, Model for network security. **Classical Encryption Techniques:** Symmetric cipher model, Substitution Techniques, Transposition techniques.

Unit-II	**11 Hrs.**

Block Ciphers, Data Encryption Standard and Advanced Encryption Standard: Simplified DES, Block cipher principles, DES, Differential and Linear cryptanalysis, Modes of operation. **AES:** Evaluation criteria for AES, AES Cipher-encryption, and Decryption, Data structure, Encryption round, Triple DES, Blowfish.

Unit-III	**11 Hrs.**

Public Key Cryptography and Key Management: Principles of public key Cryptosystem, RSA Algorithm, Key management, Diffie Hellman key exchange, Elliptic curve cryptography. **IP Security:** IP security overview, IP security architecture: Authentication, Header, Encapsulating security payload, Combining security association, Key management.

Unit-IV	**10 Hrs.**

Web Security: Web security considerations, Secure socket Layer (SSL) and Transport layer security (TLS), Secure electronic transaction (SET). **System Security:** Intruders, Viruses, and Threats.

Reference Book:

1. William Stallings, "Cryptography and Network Security Principles and Practice," 6thEdition, 2014,Pearson Education Inc., ISBN: 978-93-325-1877-3.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK2, WK3, WK4)	2 (WK2, WK3)	1 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4)	3	1	2
CO2	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	2 (WK5)	2 (WK8)	–	–	–	–	–	–	1 (WK4, WK6, WK8)	2	1	2
CO3	3 (WK2, WK3, WK4)	3 (WK2, WK3, WK4)	3 (WK5)	3 (WK8)	–	–	–	–	–	–	–	3	2	3
CO4	3 (WK2, WK3, WK4)	2 (WK2, WK3, WK4)	2 (WK5)	3 (WK8)	–	–	–	–	–	–	2 (WK4, WK6, WK8)	3	2	3

Course Code		:	**1BUEC705E**	**Compiler Design**	**Semester**	:	**07**
L:T:P		:	**3:0:0**		**Course Type**	:	**PEC**
Hours/Sem.	**Teaching**	:	**42 Hrs.**		**CIE Marks**	:	**50**
	Learning (TW+SL)	:	**42 Hrs.**		**SEE Marks**	:	**50**
	Exam	:	**06 Hrs.**		**Total Marks**	:	**100**
	Total Hrs.	:	**90 Hrs.**		**Credits**	:	**03**

Professional Competency:

Demonstrate the ability to apply the principles of automata theory, syntax analysis, and translation schemas to design, analyze, implement, and optimize the various phases of a compiler for high-level programming languages

Course Outcomes:

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

CO1: Apply the principles to design and implement a lexical analyzer.

CO2: Analyze various grammar constructs and apply top-down and bottom-up parsing techniques to construct a syntax analyzer.

CO3: Evaluate syntax-directed definitions and analyze run-time storage allocation strategies for target code execution.

CO4: Generate intermediate representations of code and apply optimization techniques to design a simple code generator..

Unit-I	**10 Hrs.**

Introduction to Compilers : Language Processors, The Structure of a Compiler (Phases of a compiler), The Evolution of Programming Languages, The Science of Building a Compiler, Applications of Compiler Technology, Programming Language Basics. **Lexical Analysis**: The Role of the Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata.

Unit-II	**11 Hrs.**

Basic Parsing Techniques and Top-Down Parsing: The Role of the Parser, Context-Free Grammars, Writing a Grammar. Top-Down Parsing: Recursive-Descent Parsing, FIRST and FOLLOW, LL (1) Grammars, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing. **Bottom-Up Parsing**: Shift-Reduce Parsing, Introduction to LR Parsing: Simple LR (SLR). More Powerful LR Parsers: Canonical LR (1) Items, LALR Parsers. Using Ambiguous Grammars, The Parser Generator Yacc

Unit-III	**1 Hrs.**

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes. **Run-Time Environments**: Storage Organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection.

Unit-IV	**10 Hrs.**

Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Back-patching, Switch-Statements, Intermediate Code for Procedures. **Code Generation and Machine-Independent Optimization**: Issues in

the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization. The Principal Sources of Optimization, Introduction to Data-Flow Analysis.

Reference Book:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, "Compilers: Principles, Techniques and Tools", 2nd Edition, Pearson Education. (Core text for all units)
2. Kenneth C. Louden, "Compiler Construction Principles & Practice", Cengage Learning.
3. Santanu Chattopadhyay, "Compiler Design", PHI Learning.
4. Tremblay, Sorenson, *"The Theory and Practice of Compiler Writing".*
5. Leland L. Beck, "System Software", 3rd Edition, Pearson Education.

CO-PO Mapping:

Course Outcomes	Programme Outcomes											PSOs		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3 (WK3, WK4)	1 (WK3, WK4)	–	–	2 (WK6)	–	–	–	–	–	–	3	2	–
CO2	3 (WK3, WK4)	2 (WK3, WK4)	2 (WK5)	–	3 (WK2, WK6)	–	–	–	–	–	–	3	3	–
CO3	2 (WK3, WK4)	3 (WK3, WK4)	3 (WK5)	1 (WK8)	2 (WK2, WK6)	–	–	–	–	–	1 (WK8)	3	3	2
CO4	2 (WK3, WK4)	2 (WK3, WK4)	3 (WK5)	2 (WK8)	3 (WK2, WK6)	–	–	–	–	–	2 (WK8)	3	3	3