COURSE STRUCTURE

COMPUTER SCIENCE & ENGINEERING

Specialization

in

Artificial Intelligence and Machine Learning

Under

Choice Based Credit System (CBCS)

Vision

To impart quality education in the field of computer science and engineering, with a specialization in artificial intelligence and machine learning, using contemporary research to meet the growing needs of industry and society.

Mission

M1: To disseminate quality education by inculcating problem analyzing and solving skills to become successful professionals.

M2: To promote artificial intelligence (AI) research that caters to the needs of industries and society.

M3: To imbibe organizational integrity and professional ethics to develop good human beings.

Program Educational Objectives (PEOs)

PEO1: Become globally competent AI professionals, researchers or entrepreneurs, for developing sustainable solutions.

PEO2: Attain positions of AI leadership in an organization and /or on teams.

PEO3: Engage in lifelong learning to improve their professional skills and knowledge to address industrial and societal needs using latest technologies.

Program Specific Outcomes (PSOs)

PSO1: Solve real-world problems using competency in computational logic, analytical ability, system design principles and programming skills.

PSO2: Design and develop hardware and software interfaces along with latest tools and technology to meet the needs of industry.

PSO3: Analyze the algorithmic principles, theory of computation, artificial intelligence and mathematical foundations for the modeling and design of computing systems.

PSO4: Apply knowledge to provide innovative solutions to existing problems and identify research gaps.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics and science, as well as the fundamentals of AI and machine learning (ML), to solve complex engineering problems related to AIML.

PO2: Problem Analysis: Identify, Formulate, review research literature and analyze complex engineering problems related to AIML and reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

PO3: Design/Development of solutions: Design solutions for complex engineering problems related to AIML and design system components or processes that meet the specified needs, appropriately considering public health and safety and cultural, societal, and environmental considerations.

PO4: Conduct Investigations of Complex Problems: Use research–based knowledge and research methods, including experiment design, data analysis and interpretation, and information synthesis, to provide valid conclusions.

PO5: Modern Tool Usage: Create, Select and apply appropriate techniques, resources and modern engineering and AIML tools including prediction and modeling to computer science related complex engineering activities with an understanding of the limitations.

PO6: The Engineer and Society: Apply Reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the CSE (AIML) professional engineering practice.

PO7: Environment and Sustainability: Understand the impact of the CSE(AIML) professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of engineering, and need for sustainable development

PO8: Ethics: Apply ethical principles and commit to professional ethics, responsibilities, and norms of engineering practice.

PO9: Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary Settings.

PO10: Communication: Communicate effectively about complex engineering activities with the engineering community and with society at large, such as comprehending and writing effective reports and design documentation, making effective presentations, and giving and receiving clear instructions.

PO11: Project Management and Finance: Demonstrate knowledge and understanding of the engineering management principles and apply these to one's own work as a member and leader in a team to manage projects and in multi-disciplinary environments.

PO12: Life-Long Learning: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Credits Distributions

Sr. No.	Category	No. of Credits
1	Humanities and Social Sciences (HS)	19
2	Basic Sciences (BS)	22
3	Engineering Sciences (ES)	21
4	Professional Core (PC)	53
5	Professional Elective (PE)	26
6	Open Elective (OE)	12
7	Project Work (PW)	32
8	Mandatory Non-Credit Courses (MNC) (4 Courses)	-
Total		**185**

1st Semester

S. NO.	CODE	CAT	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HRS/WK
				L	T	P	J		
1.	BMAS 0110	BS	Calculus	3	1	0	0	4	4
2.	BPHS 0100	BS	Physics For Engineers	2	1	0	0	3	3
3.	BCSG0031	ES	Programming-I	3	0	0	0	3	3
4.	BCSG0032	ES	Introduction to Artificial Intelligence	2	0	0	0	2	2
5.	BCSC0033	PC	Frontend Engineering	2	0	0	0	2	2
PRACTICALS									
1.	BPHS0801	BS	Engineering Physics Lab	0	0	2	0	1	2
2.	BCSG0831	ES	Programming-1 Lab	0	0	2	0	1	2
3.	BCSC0833	PC	Frontend Engineering Lab	0	0	4	0	2	4
4.	BELH 0811	HS	Professional Communication Lab - I	0	0	4	0	2	4
			TOTAL	**13**	**2**	**10**	**2**	**20**	**26**

2nd Semester

S. NO.	CODE	CAT	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HRS/WK
				L	T	P	J		
1.	BCSG0033	ES	Programming-II	3	0	0	0	3	3
2.	BCSC0034	PC	Database Technology	3	0	0	0	3	3
3.	BMAE 0111	BS	Introduction to Linear Algebra	3	1	0	0	4	4
4.	BCSG0034	ES	Data Engineering	3	0	0	0	3	3
5.	BCSG0035	ES	Design Thinking	3	0	0	0	3	3
PRACTICALS									
1.	BCSG0832	ES	Programming-II Lab	0	0	2	0	1	2
2.	BCSC0834	PC	Database Technology Lab	0	0	2	0	1	2
3.		PW	Industrial Certification-I	0	0	0	2	2	0
4.	BCSG0835	ES	Data Engineering Lab	0	0	2	0	1	2
4.	BELH 0812	HS	Professional Communication Lab – II	0	0	4	0	2	4
			TOTAL	**15**	**1**	**10**	**2**	**23**	**26**

3rd Semester

S. NO.	CODE	CAT	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HRS/WK
				L	T	P	J		
1.		PE	Data Visualization	2	0	0	0	2	2
2.	BCSC 0008	PC	Operating Systems	3	0	0	0	3	4
3.	BCSE0701	PC	Introduction To Machine Learning	3	0	0	0	3	3
4.		BS	Probability and Statistics	3	1	0	0	4	4
5.	BCSC0002	PC	Object Oriented Programming	3	0	0	0	3	3
6.	BCSG0153	ES	Digital Logic Design	3	0	0	0	3	3
PRACTICALS									
1.		PE	Data Visualization Lab	0	0	2	0	1	2
2.	BCSE0731	PC	Introduction To Machine Learning Lab	0	0	2	0	1	2
4	BCSC0801	PC	Object-Oriented Programming Lab	0	0	2	0	1	2
5.	BCS0752	ES	Digital Logic Design Lab	0	0	2	0	1	2
6.		PW	Industrial Certification-II	0	0	0	2	2	0
7.		PW	Project – I	0	0	0	0	2	0
8.		HS	Soft Skills – I	0	0	2	0	1	2
	BCSC0803	PC	Operating Systems Lab	0	0	2	0	1	2
			TOTAL	**17**	**2**	**10**	**2**	**28**	**29**

4TH SEMESTER

S. NO.	CODE	CAT	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HRS/WK
				L	T	P	J		
1.	BCSC1005	PC	Computer Organization	3	0	0	0	3	3
2.		PE	IoT and Robotics	3	0	0	0	3	3
3.	BCSC0006	PC	Data Structure and Algorithms	3	1	0	0	4	4
4.		PE	Full Stack using Flask and Django	3	0	0	0	3	3
5	BCSC1010	BS	Discrete Mathematics	3	1	0	0	4	4
PRACTICALS									
1.		PE	IoT and Robotics Lab	0	0	2	0	1	2
2.	BCSC0805	PC	Data Structure and Algorithms Lab	0	0	2	0	1	2
3.		PW	Project-II	0	0	0	2	2	0
4.	BCSC0804	PC	Computer Organization Lab	0	0	2	0	1	2
5.		PE	Full Stack Lab	0	0	2	0	1	2
6.		HS	Soft Skills-II	0	0	2	0	1	2
			TOTAL	**15**	**2**	**10**		**25**	**27**

Humanities and Social Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACT HR/WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.		Skill Enhancement – I	2	0	0	0	2		
2.		Skill Enhancement – II	2	0	0	0	2		
3.		Management	3	0	0	0	3		
PRACTICALS									
1.		Professional Communication Lab– I	0	0	4	0	2		
2.		Professional Communication Lab – II	0	0	4	0	2		
3.		Soft Skills – I	0	0	2	0	1		
4.		Soft Skills – II	0	0	2	0	1		
5.		Soft Skills – III	0	0	6	0	3		
6.		Soft Skills – IV	0	0	6	0	3		
TOTAL			**7**	**0**	**24**	**0**	**19**		

Basic Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.		Calculus	3	1	0	0	4	4	
2.		Introduction to Linear Algebra	3	1	0	0	4	4	
3.		Probability and Statistics	3	1	0	0	4	4	
4.		Discrete Mathematics	3	1	0	0	4	4	
5.		Environmental Studies	2	0	0	0	2	2	
6.		Engineering Physics	2	1	0	0	3	3	
PRACTICALS									
1.		Engineering Physics Lab	0	0	2	0	1	2	
TOTAL			**16**	**5**	**2**	**0**	**22**	**23**	

Engineering Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.		Introduction to Artificial Intelligence	2	0	0	0	2	2	
2.		Programming-I	3	0	0	0	3	3	
3.		Programming-II	3	0	0	0	3	3	
4.		Data Engineering	3	0	0	0	3	3	
5.		Design Thinking	3	0	0	0	3	3	
6.		Digital Logic Design	3	0	0	0	3	3	
PRACTICALS									
1.		Programming-I Lab	0	0	2	0	1	2	
2.		Programming-II Lab	0	0	2	0	1	2	
3.		Data Engineering Lab	0	0	2	0	1	2	
4.		Digital Logic Design Lab	0	0	2	0	1	2	
Total			**17**	**0**	**8**	**0**	**21**	**25**	

Program Core

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR / WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.	BCSC1002	Object Oriented Programming	3	0	0	0	3	3	Programming
2.	BCSC0034	Database Technology	3	0	0	0	3	3	
3.	BCSC0004	Operating Systems	3	0	0	0	3	3	
4.	BCSC1005	Computer Organization	3	0	0	0	3	3	
5.	BCSC0006	Data Structures and Algorithms	3	1	0	0	4	4	Programming
6.	BCSC0008	Computer Networks	3	1	0	0	4	4	
7.	BCSC0009	Software Engineering	3	0	0	0	3	3	
8.	BCSC0011	Theory of Automata and Formal Language	3	1	0	0	4	4	
9.	BCSC0012	Design and Analysis of Algorithms	3	0	0	0	3	3	Programming, Data Structures
10.	BCSCXXXX	Frontend Engineering	3	0	0	0	3	3	Mathematics, Programming
11.	BCSCXXXX	Introduction To Machine Learning	3	0	0	0	3	3	
12.	BCSCXXXX	Neural Networks and Deep Learning	3	0	0	0	3	3	
13.	BCSCXXXX	Natural Language Processing	3	0	0	0	3	3	
PRACTICALS									
1.	BCSC0801	Object-Oriented Programming Lab	0	0	2	0	1	2	Programming Lab
2,	BCSCXXXX	Database Technology Lab	0	0	2	0	1	2	
3.	BCSC0803	Operating Systems Lab	0	0	2	0	1	2	
4.	BCSC0804	Computer Organization Lab	0	0	2	0	1	2	

5.	BCSC0805	Data Structures and Algorithms Lab	0	0	2	0	1	2	Programming Lab
6.	BCSC0807	Design and Analysis of Algorithms Lab	0	0	2	0	1	2	Programming, Data Structures
7.	BCSCXXXX	Frontend Engineering Lab	0	0	2	0	1	2	
8.	BCSCXXXX	Natural Language Processing Lab	0	0	2	0	1	2	Programming
9.	BCSCXXXX	Introduction To Machine Learning lab	0	0	2	0	1	2	
10.	BCSCXXXX	Neural Networks and Deep Learning Lab	0	0	4	0	2	4	
		Total	39	3	22	0	53	64	

PROFESSIONAL ELECTIVES

S. NO.	CODE	SUBJECT	TEACHING SCHEME L	CREDITS T	P	J	J	CONTACTS HR/WK	PRE-REQUISITES	PRE-REQUISITES
		THEORY								
1.	Track Core		Cloud Computing	3	0	0	0	3	3	
2.			Introduction to DevOps	3	0	0	0	3	3	
3.			Generative AI	3	0	0	0	3	3	
4.	Track Electives		System Design	3	0	0	0	3	3	
			Data Visualization	2	0	0	0	2	2	
5.			LLMs, Vector Stores, and Prompt Engineering	3	0	0	0	3	3	
6.			Introduction to MLOps	3	0	0	0	3	3	
7.			Reinforcement Learning	3	0	0	0	3	3	
8.			Graph Neural Network	3	0	0	0	3	3	
9.			Adversarial Learning	3	0	0	0	3	3	

10.			Explainable AI	3	0	0	0	3	3	
11.			Edge AI and IoT	3	0	0	0	3	3	
12.			Responsible AI, Ethics in AI	3	0	0	0	3	3	
13.			AI in BioInformatics	3	0	0	0	3	3	
14.			Big Data Analytics	3	0	0	0	3	3	
15.			Time Series Analysis	3	0	0	0	3	3	
16.			Full Stack using Flask and Django	3	0	0	0	3	3	
17.			Full Stack using MERN	3	0	0	0	3	3	
18.			Full Stack	3	0	0	0	3	3	
19.			Full Stack using Server Side	3	0	0	0	3	3	
20.			IoT and Robotics	3	0	0	0	3	3	
21.			Cyber Security and Network Security Fundamentals	3	0	0	0	3	3	
22.			BlockChain	3	0	0	0	3	3	
23.			Image Processing	3	0	0	0	3	3	
24.	Domain Specific Applications		AI in Healthcare	3	0	0	0	3	3	
25.			AI in Finance	3	0	0	0	3	3	
26.			AI in Agriculture	3	0	0	0	3	3	
27.			AI in Business	3	0	0	0	3	3	
28.			AI in Banking	3	0	0	0	3	3	
29.			AI in Retail	3	0	0	0	3	3	
30.			AI in Education	3	0	0	0	3	3	
PRACTICALS										
1.	Track Electives		MLOps Lab	0	0	2	0	1	2	
2.			Full Stack Lab	0	0	2	0	1	2	

3.			Generative AI Lab	0	0	2	0	1	2	
4.			IoT and Robotics Lab	0	0	2	0	1	2	
5.			Image Processing Lab	0	0	2	0	1	2	
6.			Data Visualization Lab	0	0	2	0	1	2	
Total				-	-	-	-	26 *	-	

*Electives can be selected from this bucket making total of 26 credits

Open Electives

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.	BCSC 00xx	Open Elective-I	3	0	0	0	3	3	
2.	BCSC 00xx	Open Elective -II	3	0	0	0	3	3	
3.	BCSC 00xx	Open Elective -III	3	0	0	0	3	3	
4.	BCSC 00xx	Open Elective -IV	3	0	0	0	3	3	
Total			**12**	**0**	**0**	**0**	**12**	**12**	

Projects

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE- REQUISITES
			L	T	P	J			
1.		Project – I	0	0	0	0	2	0	
2.		Project – II	0	0	0	0	2	0	
3.		Project – III	0	0	0	0	4	0	
4.		Project – IV	0	0	0	0	4	0	
5.		Industrial Certifications - I	0	0	0	2	2	0	
6.		Industrial Certifications -II	0	0	0	2	2	0	
7.		Industrial Certifications -III	0	0	0	2	2	0	
8.		Industrial Certifications -IV	0	0	0	2	2	0	
9.		Industrial certification-V	0	0	0	2	2	0	
10.		Major Project - I	0	0	0	0	4	0	
11.		Major Project - II	0	0	0	0	4	0	
12.		Industrial Training	0	0	0	0	2	0	
		TOTAL	**0**	**0**	**0**	**16**	**32**	**0**	

Mandatory Non-Graded Course

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACT HR/WK	PRE- REQUISITES
			L	T	P	J			
1.	BCSC 00xx	MNG-I	2	0	0	0	0	2	
2.	BCSC 00xx	MNG-II	2	0	0	0	0	2	
3.	BCSC 00xx	MNG-III	2	0	0	0	0	2	
4.	BCSC 00xx	MNG-IV	2	0	0	0	0	2	
TOTAL			**8**	**0**	**0**	**0**	**0**	**8**	

SEMESTER-WISE STRUCTURE AND CURRICULUM

SEMESTER -I
SYLLABUS

BMAS 0110: CALCULUS

Course Objectives: *The aim of the course is to build knowledge and understanding of Linear Algebra among the students. The course seeks to give detailed knowledge about the applications in various fields by instilling them basic ideas about Linear Algebra.*

Prerequisites: 12th Mathematics

Credits: 04 **L–T–P-J: 3–1–0-0**

Module No.	Contents	Teaching Hours (Approx.)
I	[Course Outcome(s) No.: 1, 2, 3, 4 and 5] Differential Calculus: Introduction to partial Derivatives, Euler's theorem for homogeneous functions, Composite functions, Total derivatives, Expansion of function of several variables by Taylor's series, Jacobian and its applications, Functional dependence, Extrema of functions of several variables, Lagrange's method of multipliers. Vector Calculus: Introduction, Scalar and vector point functions, Gradient of a scalar field, Directional derivative, divergence and curl of vector field, Physical interpretation of gradient, divergence and curl, Vector Identities, Applications.	20
II	[Course Outcome(s) No.: 6, 7, 8, 9 and 10] Integral Calculus: Beta and Gamma functions, Double Integral, Change of order of integration, Triple integral, Errors and their analysis, Types of error, Numerical integration by Trapezoidal and Simpson's rules (without proof). Ordinary Differential Equations: Linear differential equation of nth order with constant coefficients, Euler-Cauchy Equations, Simultaneous differential equations, Numerical solution of first order initial value problems by Runge-Kutta IV order method. Applications to Engineering problems. Difference Equation: Finite difference operators, relation between operators, Introduction to difference equation, Solution of difference equation, Complementary function and particular integral.	20

Textbooks:

- Erwin Kreyszig, Advanced Engineering Mathematics, John Wiley & Sons, 2011.
- M. K. Jain, S. R. K. Iyengar and R. K. Jain, Advanced Engineering Mathematics, Narosa Publishing
- House, New Delhi, 2002.
- W. E. Boyce and R. D. Prima, Elementary Differential Equations, John Wiley & Sons, 2009.
- B. S. Grewal, Higher Engineering Mathematics, Khanna Publishers, New Delhi, 2014.
- S. S. Sastry, Introductory Methods of Numerical Analysis, PHI, 2012.

Reference Books:

- T. M. Apostol, Calculus, Volume I, John Wiley & Sons, Inc., USA, 1967.
- T. M. Apostol, Calculus, Volume II, Xerox Corporation, USA, 1969.
- N. P. Bali and M. Goyal, A Text Book of Engineering Mathematics, Laxmi Publications, Delhi, 2014.
- G. B. Thomas and R. Finney, Calculus and Analytic geometry, Addison Wesley, USA, 1995.

Focus: This course focuses on employability and skill development aligned with all CO's.

Employability: Prepares students for roles like data scientist, ML engineers and data analytics.

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

After studying these topics, the students will be able to:
CO1: Understand partial differentiation and its applications
CO2: Expand a real valued function of several variables in Taylor's series
CO3: Calculate Jacobian and know its applications
CO4: Use Lagrange's method of multipliers in determining the extrema of functions
CO5: Find gradient, divergence and curl of a field and know their applications
CO6: Evaluate double & triple integrals and study their applications
CO7: Apply numerical integration to solve definite integrals
CO8: Solve ordinary differential equations of higher order
CO9: Numerically solve the first order IVP by Runge-Kutta IV order method
CO10: Solve a difference equation.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

COs	POs	PSOs
CO1	PO1, PO2, PO4	PSO1
CO2	PO1, PO2	PSO1
CO3	PO1, PO2	PSO3
CO4	PO2, PO3	PSO1, PSO3
CO5	PO1, PO2, PO4, PO7	PSO1, PSO4
CO6	PO1, PO2, PO4	PSO1
CO7	PO1, PO2	PSO3
CO8	PO1, PO2, PO3	PSO1, PSO3
CO9	PO2, PO4, PO5	PSO1, PSO3
CO10	PO1, PO3, PO4, PO5	PSO1, PSO2

BPHS 0100: PHYSICS FOR ENGINEERS

Course objective: *This syllabus is designed and styled specially to give B. Tech. I year students a sound base in fundamental physics and to give them exposure to a wide range of its utility in engineering and technology.*

Prerequisites: 12th Physics

Credits:03 **L-T-P-J:2-1-0-0**

Module No.	Content	Teaching Hours
I	**Physical Optics:** Qualitative description of superposition of waves, Interference by division of amplitude, Interference in thin films, Newton's rings, Applications: Antireflection coatings and filters. Diffraction: Diffraction due to single slit, Qualitative description of N slit diffraction. Basic concepts of polarization, Optical activity and Specific rotation. **Quantum Mechanics:** Wave - particle duality, De- Broglie Hypothesis, Concept of wave packet, Phase & Group velocity, Heisensberg's Uncertainty principle & its application (Nonexistence of electron in the nucleus of atom), Wave function and its significance, Schrodinger's wave equation (Time independent), Application: Particle in one dimensional infinite potential box.	16
II	**Electromagnetics:** Maxwell's equations in differential and integral forms, Physical significance of Maxwell's equations, Concept of displacement current, Equation of continuity, Propagation of Electromagnetic waves in free space. **Laser & Fiber Optics:** Spontaneous and stimulated emissions, Population inversion, Principle of laser, Ruby laser, Application: Holography, Principle of optical fiber, Single and multi-mode fibers, Acceptance angle, Acceptance cone, Numerical aperture, Application of optical fiber in communication. **Introduction to Quantum Computing:** Basics of quantum computing, Limitations of classical computing. Difference between classical and quantum computing, Concept of Qubit and its properties, Representation of qubit by Bloch sphere, Single and two qubits & applications.	16

Reference Books/ Text Books / Cases:

1. Optics - Ajoy Ghatak (Tata Mc-Graw Hill)
2. Optics - Brijlal & Subramaniam (S. Chand)
3. Optical Fibre & Laser - Anuradha De. (New Age)
4. Fundamental of Physics - Resnick, Halliday & Walker (Wiely)
5. Concept of Modern Physics - by Beiser (Tata Mc-Graw Hill)
6. Quantum Computing and Quantum Information - by Michael A. Nielson, Isaac L. Chuang

Course Outcomes:

- CO1: Understand phenomena of Interference, Diffraction, Variation of intensities in them, and their
- applications in daily life.
- CO2: Discuss polarization of Light wave, double refraction, production and analysis of different polarized light waves and optical activity.

- CO3: Understand fundamentals of Quantum mechanics, Schrodinger's wave equations to deal with
- physics problem.
- CO4:Familiar with Maxwell equations and use them to study the Propagation of E-M waves in free
- space and conducting medium
- CO5:Understand the principle and working of Lasers.
- CO6:Familiar with mechanism of communication through Optical Fibre Cables and signal losses.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

Cos	POs	PSOs
CO1	PO1, PO2, PO4, PO12	PSO1, PSO3
CO2	PO1, PO2, PO3, PO4	PSO1
CO3	PO1, PO2, PO4	PSO3
CO4	PO1, PO3, PO5	PSO2, PSO4
CO5	PO1, PO3, PO7	PSO4
CO6	PO1, PO2, PO4, PO6, PO7	PSO4

BPHS0801: ENGINEERING PHYSICS LAB

Objective: Experiments based on theoretical aspects are set up in the laboratory to give B.Tech Students sound practical knowledge in fundamental and applied physics.

CREDIT:01 **L–T–P-J: 0–0-2-0**

Module No.	Content	Teaching Hours
I	**Group -A** 1. To determine the wavelength of monochromatic light by Newton's rings. 2. To determine the wavelength of monochromatic light with the help of Fresnel's biprism. 3. To determine the specific rotation of cane sugar solution using polarimeter. 4. To determine the wavelength of spectral lines using plane transmission /diffraction grating. 5. To determine the wave length of laser light by diffraction grating method. 6. To verify Stefan's law by electrical method. 7. To determine high resistance by leakage method using digital D.C. microvoltmeter. **Group – B** 8. To determine the specific resistance of the material of a given wire using Carey Foster's bridge. 9. To study the variation of magnetic field along the axis of current carrying circular coil and then to estimate the radius of the coil. 10. To calibrate the given ammeter by potentiometer. 11. To calibrate the given voltmeter by potentiometer. 12. To study the Hall Effect and determine Hall coefficient, carrier density and mobility of a given semiconductor. 13. To determine the energy band gap of a given semiconductor material. 14. Measurement of resistivity of semiconductor by four probe method at different temperatures and determination of the band gap.	20

Outcome: After completing the lab exercise:

- CO1: B.Tech. Students learn about the applications of basic-laws, principles and concepts of Engineering Physics in the various fields of science and technology.
- CO2: Practical training of a student reinforces his/her theoretical knowledge adequately and make him confident in subject.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

COs	POs/PSOs
CO1	PO1, PO2, PO4, PO12, PSO1, PSO3
CO2	PO1, PO2, PO3, PO4, POS1

BCSG0031: PROGRAMMING-I

Objective: *To impart adequate knowledge of problem-solving techniques and develop programming skills to implements applications using the concepts of C Language. Also, by learning the programming constructs they can easily switch over to any other language in future.*

Prerequisites: Not required

CREDITS:03 **L-T-P-J: 3-0-0-0**

Module No.	Content	Teaching Hours
I	**Generation of Programming Languages:** Low, Assembly, High and 4GL. **Language Processors:** Compiler, Interpreter, Assembler, Linker and Loader. **Algorithm:** Introduction, Features, Different Ways of stating Algorithms. **Flow Chart:** Introduction, Standard, Guidelines, Advantages and Limitations of using Flowcharts. **Basics of C:** Overview, Structure of a C program, Identifier, Keywords, Variables, Data types, Formatted Input and output. **Operators and Expression:** Assignment, Unary, Arithmetic, Relational, Logical, Bitwise, Conditional, Special operators and their precedence & Associativity. IEEE representation of data types like float & double, Lvalue and Rvalue **Type Conversion:** Type Promotion in expression, Conversion by Assignment, Truncation and Casting Arithmetic expression. **Decision and Case Control Structure:** if, if-else, nested if-else, Decisions using switch, switch versus if-else ladder, goto. **Loop Control Structure:** For loop, while loop, do-while loop, nesting of loops, break, and continue. **Arrays:** Introduction, one-dimensional and two-dimensional Array-Declaration, Initialization, Address Calculation. **Operations on Arrays:** Insertion, Deletion, Linear Search & Bubble Sort. **String:** Introduction, One dimensional and two dimensional Array-Declarations, Initialization **Operations on String:** Length, Copy, Reverse, Concatenate, Compare with & without built-in functions.	20
II	**Functions:** Declaration and Definition, Category of Functions, Parameter Passing Techniques – Call by Value, Passing Arrays to Functions. **Introduction to Storage Classes:** Auto, Static, Extern and Register. **Recursion:** Mechanics of Recursive Call, Implementation of Recursion, Recursion vs. Iteration. **The C Preprocessor:** Introduction, Macro Expansion and File Inclusion, Conditional Compilation and Miscellaneous Directives. **Pointers:** Declaration and Initialization of Pointer Variables, Accessing a Variable through its Pointer, Arrays and Pointers, Pointer and Strings, Pointer Arithmetic, Pointers to Pointers, Array of Pointers, Pointer to an Array, Two Dimensional Array and Pointers, Pointers to Functions, Dynamic Memory Allocation, void Pointer and Null Pointer. **User Defined Types:** enum, typedef, Union and Structure - Declaration, Initialization, Nested Structures, Arrays of Structures, Structure and Pointer, Passing Structure Through Function. Difference between Structures and Union. **File Handling:** Data and Information, File Concepts, File Organization, File Operations: Open, Read, and Close, Trouble in Opening a File. File Opening Modes, Working with Text Files. Random Access to Files of Records. Introduction to Command Line Arguments.	20

Text Books:

- Behrouz A. Forouzan and Richard F. Gilberg, "Computer Science – A Structured Programming Approach Using C", C Language Learning, 2007

Reference Books:

- Herbert Schildt, "C: The Complete Reference", 5th Edition, McGraw Hill Education
- K. N. King, "C Programming a Modern Approach", W. W. Norton, 2nd Edition, 2008.
- Kernighan and Ritche, "The C Programming Language", PHI, 2ndEdition, 2011.
- P. Dey and M. Ghosh, "Programming in C", Oxford University Press 2nd Edition, 2013.

Focus: This Course focuses on Employability under CO2, CO3,CO7.

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

- CO1: Understand the basic concepts of problem solving skills.
- CO2: Apply the basic principles of programming in C language.
- CO3: Understand the concepts of arrays and strings in C language.
- CO4: Apply the concepts of functions to solve real world problems.
- CO5: Illustrate the concepts of recursion.
- CO6: Understand the concepts of pointers in C language.
- CO7: Understand the basic concepts of file handling.
- CO8: Develop algorithmic solutions to simple computational problems.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO4,PO12/PSO1,PSO3
CO2	PO1,PO2,PO3,PO10/PSO1,PSO3
CO3	PO1,PO2,PO3,PO4/PSO1,PSO3
CO4	PO1,PO3, PO12/PSO1,PSO2
CO5	PO1,PO2,PO4 /PSO1,PSO3
CO6	PO1,PO2,PO3,PO4/PSO1,PSO2
CO7	PO1,PO3,PO6 /PSO1
CO8	PO1,PO2,PO4,PO10,PO12/PSO1,PSO3

BCSG0831: PROGRAMMING-I LAB

Objective: *The objective is to provide a comprehensive study of the C programming language. It stress the strengths of C, which provide students with the means of writing efficient, maintainable, and portable code.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Lab Hours
I & II	• Mapping of flow chart, Algorithm, Language • Simple C-program execution • Programs based on various operators • Programs based on Decision and case Control Structure • Programs based on Loop Control Structure • Program based on special control statement ➢ break ➢ continue • Programs based on Array Insertion, Deletion, Linear Search & Bubble Sort • Programs based on String ➢ Length, Copy, Reverse, Concatenate, Compare with & without built-in functions • Programs based on Functions. • Programs based on Storage Class. • Programs based on Recursion. • Programs based on Preprocessor. • Programs based on Pointers • Programs based on array • Programs based on string • Programs based on call by value and call by reference • Programs based on Dynamic Memory Allocation • Programs based on User Defined Data types ➢ Structure and Union ➢ Enum and Typedef • Programs based on File handling ➢ Opening a file ➢ Reading, writing and appending a file ➢ Closing file ➢ Random Access to Files of Records • Programs based on Command Line Argument.	14

Text Books:

- Behrouz A. Forouzan and Richard F. Gilberg, "Computer Science – A Structured Programming Approach Using C", C Language Learning, 2007

Reference Books:

- Herbert Schildt, "C: The Complete Reference", 5th Edition, McGraw Hill Education
- K. N. King, "C Programming a Modern Approach", W. W. Norton, 2nd Edition, 2008.
- Kernighan and Ritche, "The C Programming Language", PHI, 2ndEdition, 2011.
- P. Dey and M. Ghosh, "Programming in C", Oxford University Press 2nd Edition, 2013.

Focus: This Course focuses on Employability under CO1,CO2,CO4

Outcome: On Completion of this course, students are able to:

- CO1: Design programs involving decision structures, loops and functions.
- CO2: Understand the concepts of functions, recursion, pointers and file handling.
- CO3: Design programs involving structures, union and functions.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1,PO3/PSO1,PSO2
CO2	PO3,PO4/PSO1
CO3	PO3/PSO2,PSO4

BCSG0032: INTRODUCTION TO ARTIFICIAL INTELLIGENCE

Course Objectives: *This course is designed for students to understand the basic concepts of artificial intelligence and data analytics.*

Prerequisites: Not required

CREDITS:02 **L-T-P-J: 2-0-0-0**

Module No.	Content	Hours
I	**Introduction to Artificial Intelligence:** Overview of AI: Definition, History, and Applications. **AI Agents:** Types of Agents, Agent Environments, PEAS (Performance measure, Environment, Actuators, Sensors). **Search Algorithms:** State space search, Breadth-First Search (BFS), Depth-First Search (DFS), A* Search, Heuristics, Greedy Search, Adversarial Search. Knowledge Representation: Propositional Logic, First-Order Predicate Logic, Knowledge-based Agents. **Machine Learning Basics:** Supervised Learning: Regression, Classification. Unsupervised Learning: Clustering, Dimensionality Reduction. **Reinforcement Learning:** Concepts and Applications. Evaluation of Learning Models: Training and Testing, Overfitting, Underfitting, Cross-validation.	12
II	**Data Analytics for AI:** Introduction to Data Analytics: Importance, Types of Data, Data Preprocessing (Cleaning, Transformation, Integration). **Exploratory Data Analysis (EDA):** Descriptive Statistics, Data Visualization Techniques (Histograms, Boxplots, Scatterplots). **Statistical Analysis:** Mean, Median, Mode, Variance, Standard Deviation, Correlation, t-tests, ANOVA. **Applications of AI:** AI in Healthcare, Finance, Business, and Manufacturing. **Case Studies:** Real-world examples illustrating the application of AI in various industries. **Ethics in AI:** Ethical considerations, Bias in AI, Data Privacy.	12

Text Books:

- Artificial Intelligence: A Modern Approach, S Russel and P Norvig, 3rd Edition, 2015 Prentice Hall
- "Data Science for Business" by Foster Provost and Tom Fawcett

References Books:

- Introduction to Artificial Intelligence and Expert Systems, Dan W. Patterson, Pearson Education.
- Artificial Intelligence and Expert Systems — Patterson PHI.
- Data Analytics Made Accessible: 2020 Edition, Anil K. Maheshwari.
- Python Machine Learning by Sebastian Raschka and Vahid Mirjalili.

Course Outcome: After the completion of the course, the student will:

- CO1: Gain an understanding of the fundamental concepts and history of AI.
- CO2: Learn different types of AI agents and their environments.
- CO3: Apply search algorithms to solve basic AI problems.
- CO4: Understand and implement basic machine learning algorithms.
- CO5: Analyze data using statistical methods and visualization techniques.
- CO6: Explore real-world AI applications and understand ethical considerations in AI.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1, PO2 / PSO1
CO2	PO3, PO4 / PSO2
CO3	PO5, PO6 / PSO3
CO4	PO1, PO3 / PSO1
CO5	PO2, PO4 / PSO2
CO6	PO7, PO8 / PSO4

BCSC0033: FRONTEND ENGINEERING

Objective: *This course equip learners with foundational knowledge and practical skills in client-server architecture and front-end web development using HTML, CSS, Bootstrap, JavaScript, and modern web APIs.*

Prerequisites: Basic understanding of computer programming

Credits:02 **L-P-T-J:2-0-0-0**

Module No.	Content	Hours
I	**Introduction to Client Server Architecture**: Components of Client-Server Application, Client-Server Models and their Benefits, Static V/s Dynamic Websites. **HTML:** Introduction to HTML, formatting tags, Meta, Anchor, List, Table, Headers, Frames and iframes, Image, Form and other tags, their usage and implementation, Introduction to HTML5, Validations using HTML5. **CSS**: Introducing CSS and understanding the role of CSS, Adding CSS to our HTML Page, learning all about selectors, Learning about CSS Colors and various Color formats (hexadecimal, RGB, RGBA, and HSL format), Learning about box models in CSS and various box-model properties. Understanding CSS Units, working with layout and changing the normal flow and default layout using display, positions, and flexboxes, using Media Queries to create a responsive layout **Bootstrap :** Overview of Bootstrap , Importance and benefits of using Bootstrap, Setting up the development environment, Including Bootstrap in your project and various Bootstrap component like (Alerts , Badge , Buttons , Carousel , dropdowns , Navbar)	**12**
II	**JAVASCRIPT** : Introducing JS, Various features - Compile-Interpreted-JIT languages, Dynamically vs Statically Typed, learning all about Math objects and generating random numbers in a given range, Understanding if-else if-else if-else, Understanding while, for, for of, and for loops in JS, Ternary operator, Binary operator, Logical and Bitwise Operators, Logical AND and OR operators, Functions, Learning the first data structure - Array, and Various Array methods, Learning all about JS Objects and string **DOM**: Understanding what DOM is and how to manipulate DOM using JS, Selecting elements in a DOM tree using various selectors and Modifying the DOM **API :-** Introducing modern way of making a request using fetch web API	**12**

Reference Books:

- Chris Bates: “Web Programming-Building Internet Application”, “Wiley, India”, 2008.
- Black Book: “HTML5, CSS3, JAVASCRIPT ”, “Wiley, India”, 2ED, 2016
- Rasmus Lerdorf, Kevin Tatroe, “O’Reilly Media, Inc.", Feb 2013

Course Outcome: Upon completion of this course, the students will be able to:

- CO1: Understand the basics of client server architecture and its components.
- CO2: Explain the basics Stracture of a Web page using HTML
- CO3: understand designing of a website
- CO4: fundamental of javascript
- CO5: understand The concept of DOM

- CO6: Develop a dynamic/static website

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1, PO2, PSO1
CO2	PO1, PO2, PO3, PSO1, PSO2
CO3	PO1, PO2, PO3, PO5, PSO2
CO4	PO1, PO2, PO3, PO5, PSO2
CO5	PO1, PO2, PO3, PO5, PSO3
CO6	PO1, PO2, PO3, PO4, PO5, PSO1, PSO4

BCSC0833: FRONTEND ENGINEERING LAB

Objective: *This course introduces the building of dynamic web solutions using PHP programming and OO concepts and its connectivity with database.*

CREDITS:02 **L-T-P-J:0-0-4-0**

Module No.	Content	Lab Hours
I	Static web applications using HTML/CSS, designing web page with the help of CSS . Web applications using HTML CSS & JavaScript, Programs using Decision Control Structures, Programs using Loop Control Structures, Programs using user defined functions, Programs of Array handling and manipulation, Fetching the Real world Api and manipulation in DOM tree.	24

Reference Books:

- Chris Bates: "Web Programming-Building Internet Application", "Wiley, India", 2008.
- Black Book: "HTML5, CSS3, JAVASCRIPT ", "Wiley, India", 2ED, 2016
- Rasmus Lerdorf, Kevin Tatroe, "O'Reilly Media, Inc.", Feb 2013

Outcome: By the end of the class, students will learn to:

- CO1: Understand to solve problems with smaller Lines of Code using javascript as compared to other programming languages.
- CO2: Build dynamic web-pages with the help of HTML CSS JAVASCRIPT

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1, PO2, PO3, PSO1, PSO2
CO2	PO1, PO2, PO3, PO5, PSO2, PSO3

BELH 0811: PROFESSIONAL COMMUNICATION LAB-I

Objective: *English for Professional Communication (EPC) aims to facilitate experiential training sessions and encourage participants to develop their communication skills and sustain high-quality working relationships. In addition, this course aims to assist students develop their English language skills, prepare and present messages, stimulate critical thinking, learn organizational communication, boost confidence and overcome communication barriers. Further, each session will establish a vision for resolving key issues in communication, identify students 'potential outputs and suggest remedies for the exceptional growth.*

CREDITS:02 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I	**Grammar:** Sentences and its types Parts of Speech – Noun, Pronoun, Adjective, Verb, Adverb **Reading and Writing:** Reading Unseen Passages Reading Strategies Write Short Texts using EngagingWords	16
II	**Grammar:** Tenses- Past and Present Voice - Active and Passive **Reading and Writing:** Writing Description of Gadgets and Processes, Preparing Checklist and Questionnaire, Writing Survey Report	16

Text Books:

- Raman Meenakshi and Sharma Sangeeta, "Professional Communication" (2 nd eds.). Oxford
- University Press, 2011
- Sharma S. & Binod B., "Communication Skills for Engineers and Scientists" PHI Publications, 2015
- Peter Wentworth, Jeffrey Elkner, Allen B. Downey, and Chris Meyers, "How to Think Like a Computer Scientist: Learning with Python", 2nd edition., Open Book Project, 2012.

Reference Books:

- Leech Geoffre & Svartvik Jan. (2013). "A Communicative Grammar of English". (3 rd Eds.). Routledge Publications, 2013
- Rizvi Ashraf, "Effective Technical Communication". McGraw Hill Pvt Limited. 2015

Focus: This Course focuses on Employability under CO1,CO3.

Outcome: By the end of the class, students will learn to:

- CO1.Demonstrate critical and innovative thinking
- CO2. Display competence in Listening, Speaking, Reading and Writing
- CO3. Show an understanding of opportunities in the field of communication
- CO4. Demonstrate positive group communication exchanges

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/ PSOs
CO1	PO1 /PSO1
CO2	PO1,PO3/PSO1
CO3	PO2/PSO2
CO4	PO3/PSO1, PSO2

SEMESTER -II
SYLLABUS

BCSG0033: PROGRAMMING-II

Objective: *This course introduces the solving of mathematical problems using Python programming using Object Oriented concepts and its connectivity with database.*

Prerequisites: Not required

CREDITS:03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	Introduction to Python: Introduction and Basics; Setting up path Python Data Variables & Operators: Data Variables and its types, id () and type () functions, Coding Standards; **Control Structures:** if-else, elif, Nested if, Iteration Control structures, Break, Continue & Pass; **String Manipulation:** Accessing Strings, Basic Operations, String slices, Function and Methods. **Lists:** Introduction, accessing list, Operations, Working with lists, Function and Methods. **Tuple:** Introduction, accessing tuples, Operations, Working, Functions and Methods. **Dictionaries:** Introduction, accessing values in dictionaries, Working with dictionaries, Properties, Functions.	22
II	**Functions:** Defining & Calling a function, Passing arguments to functions – Mutable & Immutable Data Types, Different types of arguments, Recursion, Scope of variables; **Modules and Packages:** User-defined modules and Standard Library: random, numpy, scipy, sys, Math Module, String Module, List Module, Date & Time Module, Regular Expressions: match, search, replace; **Input-Output:** Printing on screen, reading data from keyboard, Opening and closing file, Reading and writing files, Functions. **Exception Handling:** Exception, Exception Handling, except clause, try? finally clause, User Defined Exceptions. Basics of Python for Data Analysis, Introduction to series and data frames& Python using Pandas.	22

Textbooks:

- Paul Barry: "Head First Python "O'Reilly Media, Inc.", 2010.

Reference Books:

- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

Focus: This Course focus on Employability under CO1, CO2,CO3,CO5,CO7.

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

- CO1: Understand the basics of Python Programming.
- CO2: Apply the concepts of control structures and string manipulations of python programming.
- CO3: Understand the use of data structures available in Python List, Tuple and Dictionary.
- CO4: Experiment user-defined functions and access built-in functions.

- CO5: Experiment user-defined modules and access built-in modules- math, random, string, date, time, date time.
- CO6: Develop the programs using the concept of File Handling.
- CO7: Develop programs based on Exceptional Handling.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO2/PSO4
CO2	PO4/PSO1
CO3	PO5/PSO4
CO4	PO5,PO7/PSO1
CO5	PO2,PO8/PSO4
CO6	PO3,PO10/PSO2
CO7	PO5,PO9/PSO1

BCSG0832: PROGRAMMING-II LAB

Objective: *This course introduces the solving of problems using Python programming using OO concepts and its connectivity with database.*

CREDITS:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Lab Hours
I & II	Programs based on the concepts of: • Building Python Modules • Obtaining user Data • Printing desired output Programs based on the concepts of: • Conditional if statements • Nested if statements • Using else if and elif Programs based on the concepts of Iteration using different kinds of loops Usage of Data Structures • Strings • Lists • Tuples • Sets • Dictionary Program based on the concepts of User-defined modules and Standard Library (random, numpy, scipy, sys, Math Module, String Module, List Module). Program based on Input Output. Program based on exception Handling. Program based on Simple Data analysis. Program based on Pandas.	20

Text Books:

- Paul Barry: "Head First Python "O'Reilly Media, Inc.", 2010.

Reference Books:

- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

Focus: This Course focuses on Employability under CO1,CO2,CO3.

Outcome: By the end of the course, students will learn to:

- CO1: Apply OO concepts using Python programming.
- CO2: Apply in-built packages defined in Python.
- CO3: Apply front-end as Python Programming to connect with any back-end.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2/PSO1
CO2	PO3/PSO4
CO3	PO5/PSO2

BCSC0034: DATABASE TECHNOLOGY

Objective: *The objective of the course is to enable students to understand and use a relational database system. Students learn how to design and create a good database.*

Prerequisites: Not required

CREDITS: 03 **L-T-P-J: 3-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** An Overview of Database Management System, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence, Database Language and Interfaces (DDL, DML, DCL), Database Development Life Cycle (DDLC) with case studies. **Data Modeling Using the Entity-Relationship Model:** ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Specialization, Generalization, Aggregation, Reduction of an ER Diagram to Tables. **Relational Data Model and Language:** Relational Data Model Concepts, Integrity Constraints: Entity Integrity, Referential Integrity, Key Constraints, Domain Constraints, **Relational Algebra:** Selection, Projection, Union, Intersection, Set Difference, Cross Product, Joins: Inner Join (Theta Join, Equi Join, Natural Join), Outer join (Left, Right, Full Outer Join), Division. **Database Design & Normalization:** Functional Dependencies, Primary Key, Foreign Key, Candidate Key, Super Key, Normal Forms, First, Second, Third Normal Forms, BCNF, 4th Normal Form, 5th Normal Form	26
II	**Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Deadlock Handling. **Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, 2PL, Time Stamping Protocols for Concurrency Control, Validation Based Protocol. **Distributed Database:** Introduction of Distributed Database, Data Fragmentation and Replication. **Database Programming using Python**: Database connectivity, Retrieving Data from Database, Parameters Passing, Execute many Methods, Cursor Attributes, Invoke Stored Procedures, Invoke Stored Functions.	26

Text Books:

- Elmasri and Navathe, "Fundamentals of Database Systems", 6th Edition, Addison Wesley, 2010.
- Sadalage, P. & Fowler , "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Pearson Education, 2012.

References Books:

- Date C J, " An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH, 1998.
- Redmond, E. &Wilson, "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement", 1st Edition.

Course Outcome: After completion of course, student will be able to:

CO1: Understand the concept of database management systems and Relational database.
CO2: Identify the various data model used in database design.
CO3: Design conceptual models of a database using ER modeling for real life applications and construct queries in Relational Algebra.
CO4: Create and populate a RDBMS for a real life application, with constraints and keys, using SQL.
CO5: Select the information from a database by formulating complex queries in SQL.

CO6: Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.
CO7: Discuss recovery system and be familiar with introduction to web database, distributed databases.
CO8:Design and implement the database system with the fundamental concepts of DBMS using Python.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1, PO2, PSO1, PSO4
CO2	PO1, PO2, PO3, PSO1, PSO4
CO3	PO1, PO2, PO3, PO4, PSO1, PSO3
CO4	PO1, PO2, PO3, PO5, PSO2, PSO3
CO5	PO1, PO2, PO3, PO5, PSO2, PSO3
CO6	PO1, PO2, PO3, PO4, PO5, PSO1, PSO3
CO7	PO2, PO4, PO5, PO12, PSO2, PSO4
CO8	PO1, PO3, PO5, PO10, PO12, PSO2, PSO4

BCSC0834: DATABASE TECHNOLOGY LAB

Objective: *The lab aims to develop an understanding of different applications and constructs of SQL, PL/SQL.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Write the SQL queries for data definition and data manipulation language. • To implement various operations on a table. • To implement various functions in SQL. • To implement restrictions on the table. • To implement the concept of the grouping of Data. • To implement the concept of Joins in SQL. • To implement the concept of sub-queries. • To implement the concept of views, sequence. • To implement the concept of Procedure function and Triggers.	12

References Books:

- Date C J, " An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH, 1998.
- Majumdar& Bhattacharya, "Database Management System", TMH

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

- CO1: Apply SQL queries for DML and DDL.
- CO2: Develop the SQL queries for real life scenarios.
- CO3: Implement the procedural language (PL/SQL) and Triggers.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

Cos	POs/PSOs
CO1	PO1, PO2/PSO1, PSO4
CO2	PO1, PO2/PSO1, PSO4
CO3	PO2, PO3, PO5/PSO2, PSO3

BCSG0034: DATA ENGINEERING

Objective: *To introduce first-year undergraduate students to the fundamental concepts, techniques, and tools used in data engineering, preparing them for handling and managing large-scale data systems.*

Prerequisites: Basic knowledge of programming, data modelling and database technology.

CREDIT: 3 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction to Data Engineering**: Overview of Data Engineering: Definition, Importance, and Applications. **Data Engineering Lifecycle:** Data Generation, Collection, Storage, Processing, and Analysis, The Role of a Data Engineer. **Data Collection**: Sources of Data, Data Lake, Batch Processing, Streaming, Web Scraping, APIs and Data Extraction. **Data Modeling**: Concepts, E-R Diagrams, Normalization and denormalization, OLAP, OLTP **Data Storage**: Relational Databases. NoSQL Databases. Data Warehouses. **Data Processing and Transformation**: ETL (Extract, Transform, Load) Processes	20
II	**Data Cleaning Techniques**- Batch Processing vs. Stream Processing **Data Integration**: Combining Data from Multiple Sources. Data Integration Tools. Data Quality and Consistency **Data Pipelines**: Designing and Implementing Data Pipelines. Workflow Orchestration Tools (e.g., Apache Airflow). Monitoring and Maintaining Data Pipelines **Data Security and Privacy**: Ensuring Data Security. Data Encryption Techniques. Compliance with Data Privacy Regulations	20

Textbooks:

- *Data Engineering with Python* by Paul Crickard.
- *Fundamentals of Data Engineering* by Joe Reis and Matt Housley.

Reference Books:

- *The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling* by Ralph Kimball and Margy Ross.
- *Designing Data-Intensive Applications* by Martin Kleppmann.
- *Big Data: Principles and Best Practices of Scalable Real-Time Data Systems* by Nathan Marz and James Warren.

Course Outcome: Upon completion of the course, students will:

- CO1: Understand the basics and importance of data engineering.
- CO2: Learn different data models and schema design techniques.
- CO3: Gain proficiency in data integration and ETL processes.
- CO4: Explore and use various big data technologies and tools.
- CO5: Apply data engineering concepts to real-world scenarios.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1, PO2 / PSO1
CO2	PO3, PO4 / PSO2
CO3	PO5, PO6 / PSO3
CO4	PO1, PO3 / PSO1
CO5	PO2, PO4 / PSO2
CO6	PO2, PO4, PO7 / PSO4

BCSG0835: DATA ENGINEERING LAB

Objective: *To provide practical experience in implementing data engineering techniques and tools using Python and other relevant technologies.*

CREDITS: 01 **L-T-P-J: 0-0-2-0**

Module No.	Content	Teaching Hours
I	1. Data modeling using ER diagram 2. Data preprocessing – Cleaning data, noise elimination, feature selection and dimension reduction 3. Schema design for OLAP and OLTP 4. ETL Pipeline implementation- Extract data from various sources. Transform data using cleaning, aggregation, and enrichment techniques. Load data into different storage solutions. 5. Building ETL Pipelines – Extracting data from API and databases 6. Data Transformation 7. Setting up AirFlow 8. Creating Directed Acyclic Graph using AirFlow 9. Data cleaning 10. Data storage using NoSQL	20

Textbooks:

- *Data Engineering with Python* by Paul Crickard.

Reference Books:

- *Designing Data-Intensive Applications* by Martin Kleppmann.
- *Python Data Science Handbook* by Jake VanderPlas.
- *Hadoop: The Definitive Guide* by Tom White.

Course Outcome: Students will be able to:

- CO1: Use SQL for database management.
- CO2: Design and implement effective database schemas.
- CO3: Perform ETL operations to manage data flows.
- CO4: Utilize big data frameworks and tools for data processing.
- CO5: Create and manage data pipelines using orchestration tools.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1, PO2 / PSO1
CO2	PO3, PO4 / PSO2
CO3	PO5, PO6 / PSO3
CO4	PO1, PO3 / PSO1
CO5	PO2, PO4 / PSO2

BCSG0035:DESIGN THINKING

Objective: *The objectives of this course are to impart knowledge on design thinking process for understanding complex designs, to provide design skills to analyze design thinking issues and apply the tools and techniques of design and to use design thinking concept in product development.*

Prerequisites: Not required

CREDITS:03 **L-T-P-J:3-0-0-0**

Module No.	Content	Hours
I	**Introduction To Design Thinking-** Design Thinking Process: Types of the thinking process, Common methods to change the human thinking process, Design thinking: Definition, Origin of design thinking, Importance of design thinking, Design vs Design thinking, Problem solving, Understanding design thinking and its process model, Design thinking tools. **Empathize-** Design thinking phases, How to empathize, Role of empathy in design thinking, purpose of empathy maps, Things to be done prior to empathy mapping, Activities during and after the session, Understanding empathy tools : Customer Journey Map, Personas. **Ideation-**Challenges in idea generation, need for systematic method to connect to user, Visualize, Empathize, and Ideate method, Importance of visualizing and empathizing before ideating, Applying the method, Ideation Tools: How Might We? (HMW), Story board, Brainstorming, MOM Test	20
II	**Implementation of Design Thinking in Problem Solving & Product Development:** Methodology in Product Development: Agile Vs Waterfall, Understanding AARRR Framework in Product Development, Root cause Analysis **Case Study:** 1. Spotify Social: https://medium.muz.li/what-if-you-can-create-listening-sessions-on-spotify-a-user-experience-design-case-study-fb8d20dd0e18 2. Product management Case Study on Notion, Medium, Slack https://zeda.io/blog/product-management-case-studies **Case Study Assignment:** 1. Improve the UI of PayTM for increasing user engagement 2. Suggest a new feature in Amazon app to increase user engagement 3. Suggest improvement in Instagram Threads to compete with its rival Twitter OR vice-verce *Case Study Competition*: Pick any one of the following for Case Study: Spotify, Threads, Instagram, Zepto, Meesho, Zomato, PolicyBazar, Groww, BigBasket, Zomato, Dunzo, Blinkit, PayTM, PhonePe	20

Text Book:

- Karmic Design Thinking by Prof. Bala Ramadurai, available at Amazon (paperback), Flipkart, halfpricebooks.in
- S. Salivahanan, S.Suresh Kumar, D.Praveen Sam, “Introduction to Design Thinking”, Tata Mc Graw Hill, First Edition, 2019.
- Kathryn McElroy, “Prototyping for Designers: Developing the best Digital and Physical Products”, O’Reilly, 2017.

Reference Books:

- Michael G. Luchs, Scott Swan, Abbie Griffin, ”Design Thinking –New Product Essentials from PDMA”, Wiley, 2015.

- Vijay Kumar, "101 Design Methods: A Structured Approach for Driving Innovation in Your Organization", 2012.
- The Design of Everyday Things by Don Norman, 2013 Available on Amazon.
- The Mom Test: How to talk to customers & learn if your business is a good idea when everyone is lying to you, Rob Fitzpatrik.

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

- CO1: Analyze design thinking concepts and principles to perform human centered design process for creative problem solving.
- CO2: Create empathy maps to visualize user attitudes and behavior for gaining insights of customers.
- CO3: Develop innovative products or services for a customer base using ideation techniques.
- CO4: Build & improve product by implementing concept of design thinking.
- CO5: Apply different methodology & framework for product development.
- CO6: Improve product features, UI/UX to increase user retention and engagement.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1, PO2 / PSO1
CO2	PO3, PO4 / PSO2
CO3	PO5, PO6 / PSO3
CO4	PO1, PO3 / PSO1
CO5	PO2, PO4 / PSO2
CO6	PO2, PO4, PO7 / PSO4

BELH 0812: PROFESSIONAL COMMUNICATION- II

Objective: *The objective of this course is to give students exposure of listening and speaking English in various communicative contexts. The course provides them with an opportunity to form their opinion, develop their thinking skills, analyze their thoughts and express themselves in an effective way and to equip them with the ability to communicate effectively in English.*

Prerequisites: BELH 0811: PROFESSIONAL COMMUNICATION- I

CREDIT: 01 **L-T-P-J: 0-0-2-0**

Module No.	Content	Teaching Hours
I	Introducing oneself: Salutation, information, change in content according to context Reading Practice to Improve Pronunciation: English Pronunciation Training, Reading newspaper articles Conducting Interviews - De Grasse Tyson: we have to believe science on climate change Relevance of Intonation : Role Play	16
II	Interpreting Visual Information: "An Irish Goodbye" Tom Berkley& Ross White Process Description: Website Design and Development Process Presenting Ideas: The Curious Effects of AI on the Artist, How AI could save (not destroy) Education by Sal Khan Drafting Advertisement: Teacher by 7-Eleven, Unsung Hero, British Airways: Fueled by Love	16

Audio-Visual Material: **Study Material of 'Orell' Digital Language Lab.**

Text Books:

- Norman Lewis, Word Power Made Easy, Goyal Publications & Distributers, Delhi.
- Mohan, Krishan & N.P. Singh, Speaking English Effectively, Macmillan India Ltd., New Delhi
- Cambridge English Pronouncing Dictionary by Daniel Jones

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

- CO1:Listen, understand and analyse a variety of speeches, documentaries and interviews.
- CO2:Express themselves with correct pronunciation and fluency.
- CO3:Use appropriate vocabulary.
- CO4: Participate in discussions on social and professional issues.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

Cos	POs/ PSOs
CO1	PO1 /PSO1
CO2	PO1,PO3/PSO1
CO3	PO2/PSO2
CO4	PO3/PSO1, PSO2

BMAS0111: INTRODUCTION TO LINEAR ALGEBRA

Objective: *The aim of the course is to build knowledge and understanding of linear algebra among the students. The course seeks to give detailed knowledge about the applications in various fields by instilling in them basic ideas about Linear Algebra.*

Prerequisites: 12th level Mathematics

CREDIT: 04 **L-T-P-J: 4-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction to Matrices:** Inverse of a matrix by elementary transformations, Rank of a matrix by reduction to echelon form. **Vector Space:** Vector equations, geometrical description of $\mathbb{R}^2$, $\mathbb{R}^3$, $\mathbb{R}^n$, Vector spaces and Subspaces, Linear combination of vectors, Linear independence and dependence of vectors, Basis and dimension of a vector space. **Linear Transformation:** Introduction, Kernel and range of linear transformation, Rank-Nullity theorem (without proof), Matrix of a linear transformation. **Partitioned Matrices:** Addition, scalar multiplication, multiplication and inverse of partitioned matrices. Applications.	**20**
II	**Inner Product Space:** Introduction, Orthogonal and orthonormal vectors, Norm of a vector and a matrix, condition number of a matrix. Orthogonal projection, Gram-Schmidt orthogonalization process, QR factorization of a matrix. **Linear Systems:** Systems of linear equations, Solution sets of linear systems, sensitivity analysis. Applications in network flow. **Eigen values and Eigen vectors:** Characteristic equation, Eigen values and Eigen vectors, Cayley-Hamilton theorem (without proof), Diagonalization of a matrix. **Matrix Factorization:** LU factorization, Applications in Computer Graphics.	**20**

Text Books:

- D. C. Lay, Linear Algebra and its Applications, Pearson Education, 2023.
- S. Lipschutz, Schaum's Outline of Linear Algebra, Schaum's Outline Series, 2021.
- S. Kumarsena, Linear Algebra – A Geometric Approach, PHI, 2022.

Reference Books:

- K. Hoffman and R. Kunji, Linear Algebra, Pearson, 2022.
- C. D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM, 2023.

- K. M. Abadir, R. Magnus, Matrix Algebra, Cambridge University Press, 2012.

Course Outcomes: After studying these topics, the students will be able to:

- CO1: Calculate the inverse and rank of a matrix by elementary transformations.
- CO2: Know the different types of vector spaces and subspaces.
- CO3: Find the basis and dimension of a vector space.
- CO4: Understand the linear transformation and form a matrix of linear transformation.
- CO5: Partition a large matrix and apply operations.
- CO6: Know inner product space and Gram-Schmidt orthogonalization process.
- CO7: Solve a linear system and carry out its sensitivity analysis.
- CO8: Identify Eigen values, Eigen vectors and diagonalize a matrix.
- CO9: Use Cayley-Hamilton theorem.
- CO10: Factorize a matrix and know its applications.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs)

COs	POs	PSOs
CO1	PO1, PO2, PO4	PSO1
CO2	PO1, PO2	PSO1
CO3	PO1, PO2	PSO3
CO4	PO2, PO3	PSO1, PSO3
CO5	PO1, PO2, PO4, PO7	PSO1, PSO4
CO6	PO1, PO2, PO4	PSO1
CO7	PO1, PO2	PSO3
CO8	PO1, PO2, PO3	PSO1, PSO3
CO9	PO2, PO4, PO5	PSO1, PSO3
CO10	PO1, PO3, PO4, PO5	PSO1, PSO2

SEMESTER -III
SYLLABUS

BCSEXXXX: DATA VISUALIZATION

Objective: *This course enables the students to understand the principles of effective data visualization, explore various types of charts and graphs and learn best practices in data design and storytelling.*

Prerequisites: Basics of Python and AI.

CREDITS:03 **L-T-P-J:3-0-0-0**

Module No.	Contents	Teaching Hours
I	**Introduction to Data Visualization**, Importance, applications, and tools, Visual Perception and Cognitive Load, Principles of Good Visualization, Clarity, accuracy, and aesthetics: Choosing the right chart for the data, Chart Types and Use Cases, Bar, Line, Pie, Scatter, Histogram, Boxplot, Heatmap, Bubble, Area charts etc. **Advanced Visualization Techniques:** Interactive Plots with Plotly and Bokeh, Advanced Seaborn Techniques, Pairplots, Catplots, FacetGrid, and Heatmaps, Dashboards with Plotly Dash / Streamlit, Visualizing Multidimensional and Time Series Data, Geo-visualization with Folium and Plotly. Business Analytics, Health Informatics, AI/ML Model Interpretability (SHAP, Partial Dependence), Ethical and Cultural Aspects of Visualization. **Assignments & Activities:** Create visualizations for a given dataset using Matplotlib and Seaborn, Analyze misleading visualizations and redesign them; mini-project: Storytelling with a dataset (e.g., COVID-19, Olympics, GDP)	**20**
II	**Introduction to Business Intelligence (BI).** Overview of Power BI and Tableau, Installation and Interface Tour, Connecting to Data Sources, Importing data (Excel, CSV, SQL Server, Web), Data types and formats, Live vs Extract connections (Tableau), Direct Query vs Import (Power BI), Power Query Editor (Power BI), Data Interpreter & Data Prep (Tableau), Cleaning, filtering, pivoting/unpivoting, Merging and appending data sources, Understanding tables and relationships, Creating calculated columns, measures, Introduction to DAX (Power BI), Introduction to Tableau Calculated Fields,Creating core charts: Bar, Line, Pie, Scatter, Maps, Filters, slicers, and drill-down (Power BI), Quick filters, parameters, sets, groups (Tableau) Tooltips, color, size, labels, and hierarchies, **Dashboards and Storytelling** using PowerBI and Tableau, Designing dashboards and layouts, Using tiles, containers, and formatting, Actions and Interactivity (filters, URL, highlights), Dashboard best practices and storytelling with data. Publishing reports to Power BI Service, Tableau Public and Tableau Server, Exporting dashboards as PDF/images, Scheduled refresh and data update. **Activities:**Build an interactive dashboard for a public dataset; Capstone Project: Data story + dashboard on a chosen theme, Peer review and presentation sessions. Dashboard themes: Business Analytics Dashboard, Sales/Marketing Dashboard, Healthcare/COVID Dashboard, Education or HR Analytics Dashboard.	**20**

Text Book:

- Data Visualization: A Practical Introduction" by Kieran Healy
- Python Data Visualization: An Introduction to Data Visualization in Python with Matplotlib, Seaborn, and Plotly" by Adel Osmani
- The Visual Display of Quantitative Information" by Edward Tufte

Reference Books:

- "Storytelling with Data: A Data Visualization Guide for Business Professionals" By Cole Nussbaumer Knaflic
- "Data Visualization: A Practical Introduction" By Kieran Healy
- "Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures" By Claus O. Wilke
- "Interactive Data Visualization for the Web" By Scott Murray
- "Python Data Science Handbook" By Jake VanderPlas
- "Designing Data Visualizations: Representing Informational Relationships" By Noah Iliinsky and Julie Steele
- "Data Visualization with Python and JavaScript"By Kyran Dale
- "Practical Statistics for Data Scientists" By Peter Bruce, Andrew Bruce, and Peter Gedeck
- "Information Dashboard Design: The Effective Visual Communication of Data" By Stephen Few

Course Outcomes (CO): Upon completion of this course students will be able to:

- **CO1**: Understand the Fundamentals of Data Visualization
- **CO2**: Apply Visualization Best Practices to Gain Proficiency with Visualization Tools
- **CO3:** Design and Develop Data Stories and Dashboards
- **CO4:** Analyze Real-World Use Cases and Ethical Considerations.
- **CO5:** Collaborate and Present Visual Data Projects

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4
CO6	PO1/PSO2,PSO4

BCSEXXXX: DATA VISUALIZATION LAB

Objective: *The main aim of the lab enables the students to understand the principles of effective data visualization, explore various types of charts and graphs and learn best practices in data design and storytelling.*

CREDIT: 01 **L-T-P-J: 0-0-2-0**

Module No.	Content	Teaching Hours
I	• Getting Started with Power BI/Tableau. Import a simple CSV dataset (e.g., Superstore, Iris) and explore basic views: table, bar, and line charts. • Data Cleaning and Transformation. Use Power Query (Power BI) or Data Interpreter (Tableau) to clean missing values, rename columns, change data types, and create calculated fields. • Creating Bar and Line Charts for series data. Use sales or stock market datasets to create bar charts for categories and line charts for trends over time. Add tooltips and labels. • Interactive Dashboards with Filters and Slicers. Build a dashboard for a retail dataset that includes filters for region, category, and date. Use slicers (Power BI) or filter actions (Tableau). • Geo-Visualization with Maps. Use a dataset with location data (e.g., country/state/city-wise sales or COVID-19 cases) and create a filled map or symbol map. • Pie and Donut Charts for Category Proportion. Use a dataset like the Olympic medals dataset to show proportions using pie and donut charts. Analyze medal share by country. • Creating KPI Cards and Custom Metrics. Create KPIs such as Total Sales, Profit Margin, and Growth Rate using DAX (Power BI) or Calculated Fields (Tableau). Display as cards. • Heatmaps and Tree Maps. Create heatmaps for correlation matrices and tree maps for category-wise sales in a multilevel category dataset (e.g., Category > Subcategory). • Time Series Forecasting and Trend Lines. Use a stock price or sales dataset to add trend lines, moving averages, and forecasts. Explore Tableau's forecast model or Power BI's analytics panel. • Capstone: Storytelling with Dashboards. Choose a theme (e.g., GDP growth, education statistics, environmental data), prepare and clean the data, and build an interactive dashboard that tells a compelling story using both Tableau and Power BI. Present findings.	**24**

Text Book:

- Data Visualization: A Practical Introduction" by Kieran Healy
- Python Data Visualization: An Introduction to Data Visualization in Python with Matplotlib, Seaborn, and Plotly" by Adel Osmani
- The Visual Display of Quantitative Information" by Edward Tufte

Reference Books:

- "Storytelling with Data: A Data Visualization Guide for Business Professionals" By Cole Nussbaumer Knaflic
- "Data Visualization: A Practical Introduction" By Kieran Healy
- "Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures" By Claus O. Wilke

- "Interactive Data Visualization for the Web" By Scott Murray
- "Python Data Science Handbook" By Jake VanderPlas
- "Designing Data Visualizations: Representing Informational Relationships" By Noah Iliinsky and Julie Steele
- "Data Visualization with Python and JavaScript"By Kyran Dale
- "Practical Statistics for Data Scientists" By Peter Bruce, Andrew Bruce, and Peter Gedeck
- "Information Dashboard Design: The Effective Visual Communication of Data" By Stephen Few

Course Outcomes (CO): Upon completion of this course students will be able to:

- **CO1**: Understand the Fundamentals of Data Visualization
- **CO2**: Apply Visualization Best Practices to Gain Proficiency with Visualization Tools
- **CO3:** Design and Develop Data Stories and Dashboards
- **CO4:** Analyze Real-World Use Cases and Ethical Considerations.
- **CO5:** Collaborate and Present Visual Data Projects

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4

BCSC0004: OPERATING SYSTEMS

Objective: *This course aims to introduce the concept of computer operating system. In particular , focuses on basic hardware architectural issues that affects the nature nature and performance of software.*

Prerequisites: basic understanding of computers system.

Credits: 03 **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Operating System and its Classification - Batch, Interactive, Multiprogramming, Time sharing, Real Time System, Multiprocessor Systems, Multithreaded Systems, System Protection, System Calls, Reentrant Kernels, Operating System Structure- Layered structure, Monolithic and Microkernel Systems, Operating System Components, Operating System Functions and Services. **Processes:** Process Concept, Process States, Process State Transition Diagram, Process Control Block (PCB), Process Scheduling Concepts, Threads and their management. **CPU Scheduling:** Scheduling Concepts, Performance Criteria, Scheduling Algorithms, Multiprocessor Scheduling. **Process Synchronization:** Principle of Concurrency, Implementation of concurrency through fork/join and parbegin/parend, Inter Process Communication models and Schemes, Producer / Consumer Problem, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Synchronization Hardware. **Classical Problem in Concurrency:** Dining Philosopher Problem, Readers Writers Problem.	20
II	**Deadlock:** System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock, Combined Approach. **Memory Management:** Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Paging, Segmentation, Paged segmentation. **Virtual memory concepts**: Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Locality of reference. **I/O Management and Disk Scheduling:** I/O devices, I/O subsystems, I/O buffering, Disk storage and disk scheduling. **File System:** File concept, File organization and access mechanism, File directories, File allocation methods, Free space management	20

Text Books:

- Silberschatz, Galvin and Gagne, "Operating Systems Concepts",9th Edition, Wiley, 2012.

References:

- SibsankarHalder and Alex a Aravind," Operating Systems", 6th Edition, Pearson Education, 2009.
- Harvey M Dietel, "An Introduction to Operating System", 2nd Edition, Pearson Education, 2002.
- D M Dhamdhere, "Operating Systems: A Concept Based Approach", 2nd Edition, 2006.
- M. J. Bach, "Design of the Unix Operating System", PHI, 1986.

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

- CO1: Understand the classification of operating system environment.
- CO2: Understand the basic of process management.
- CO3: Apply the concept of CPU process scheduling for the given scenarios.
- CO4: Illustrate the process synchronisation and concurrency process in the operating system.
- CO5: Analyze the occurrence of deadlock in operating system.
- CO6: Describe and analyze the memory management and its allocation policies.
- CO7: Understand the concepts of disk scheduling

Mapping of course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO7/PSO1
CO2	PO1/PSO2/PSO1
CO3	PO1,PO4/PSO1,PSO3
CO4	PO4,PO3, PO6/PSO3, POS4
CO5	PO1,PO4,PO6/PSO1,PSO3
CO6	PO1,PO6/PSO1,PSO3
CO7	PO1,PO2/PSO1,PSO2
CO8	PO1,PO2, PO7/PSO1,PSO3

BCSC0803: OPERATING SYSTEMS LAB

Objective: The lab aims to develop an understanding of the operation of the UNIX operating system.

Credit:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Implement the following basic commands (with options) used in • UNIX/LINUX OS. • Write and implement the basic vi editor commands. • Shell scripts that use simple commands. • Decision based Shell scripts. • Shell scripts related to strings. • Shell scripts using pipes. • Shell scripts with loop statements. • Demonstration and solution for race condition. • Demonstration and use of System Calls. • Implement the basics of IPC in UNIX.	20

Text Book:

- SibsankarHalder and Alex a Aravind," Operating Systems", 6th Edition, Pearson Education, 2009.
- Harvey M Dietel, "An Introduction to Operating System", 2nd Edition, Pearson Education, 2002.
- D M Dhamdhere, "Operating Systems: A Concept Based Approach", 2nd Edition, 2006.

Reference Books:

- M. J. Bach. , "Design of the Unix Operating System", PHI, 1986.

Course Outcome: After completion of Lab, student will be able to:

- CO1: Implement the basic operations on UNIX operating systems.
- CO2: Demonstrate the working of systems calls.
- CO3: Demonstrate message passing in Unix operating system.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO4/PSO1
CO2	PO2,PO3/PSO1,PSO1
CO3	PO1,PO4, PO5/,PSO1,PSO2

BCSE0701: INTRODUCTION TO MACHINE LEARNING

Objective: *The objective of this course is to introduce the basic process of machine learning, mathematical modelling of supervised and unsupervised machine learning methods and to utilise the combined voting of the different machine learning methods for solving real-world problems using machine learning approaches.*

Prerequisites: Basics of AI, visualization, and data handling

Credit-03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Introduction to machine learning, Applications, and motivation, programming approach vs. machine learning approach in Artificial Intelligence, components of a learning problem (such as data, model, and error functions), basic learner, types of learning, features and feature vector, process of learning (training), testing, bias and variance error. **Python for Data Science**-Numpy, Pandas for preprocessing, Matplotlib and Jupyter Notebook. **Data Preprocessing**- Importing the Libraries, Importing the dataset, data imputation, Encoding Categorical Data, Splitting the dataset into Training and Test set, Feature Scaling. **Forecasting and Learning Theory:** Predicting numerical values: linear and non-linear regression. (Implementation on any real-world dataset e.g. Boston Housing), Regression model using Gradient Descendent. **Validation:** True and sample error, over-fitting, role of cross validation, regularization, bias-variance analysis. **Performance Measures:** Types-of-errors, accuracy, confusion-matrix, precision-recall. **Project:** Estimation of diabeties using regression with gradient descendent.	21
II	**Dimensionality Reduction:** Feature Selection vs. feature extraction, Principal Component Analysis (PCA), Singular Value Decomposition. **Supervised Learning:** support vector machine, decision tree, Naïve Bayes classifier. **Unsupervised Learning:** clustering, Hierarchical clustering **Ensemble Learning**: Introduction, Bagging, Boosting, Improving classification, Ada-Boost algorithm. **Machine learning Approach in NLP-** Introduction to NLP libraries e.g. spacy, NLTK**.** Text classification using spacy, sentiment classification using spacy on IMDB dataset. **Introduction of CNN-** Difference between ANN and CNN, libraries to implement CNN and designing an application of image processing using CNN.	21

Text Books:

- Alpaydin, E. . Introduction to machine learning. MIT press, 2009.
- Bishop, C. M. . Pattern recognition and machine learning (information science and statistics) springer-verlag new york. Inc. Secaucus, NJ, USA, 2006.

Reference Books:

- Harrington, P. . Machine learning in action. Shelter Island, NY: Manning Publications Co , 2012.

Course Outcome: After completion of Lab, student will be able to:

- CO1: Apply the basic concepts of machine learning including bias-variance tradeoff.
- CO2: Apply the concepts of regression.
- CO3: Conceptualize supervise and re-enforcement learning for classification.

- CO4: Formulate the ensemble methods for improving classification.
- CO5: Apply ANN with optimization in machine learning.
- CO6: Design and develop projects based on machine learning.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4
CO6	PO1/PSO2,PSO4

BCSE0731: MACHINE LEARNING LAB

Objective: This course aims *to introduce the basic process of machine learning, mathematical modelling of supervised and unsupervised machine learning methods, and to utilize the combined voting of the different machine learning methods for solving real-world problems using machine learning. approach.*

Credit:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Introduction to python, numpy and pandas • Implementation of Simple linear regression • Implementation of Multiple linear regression • Implementation of logistic regression • Implementation of regression models with regularization • Implementation of dimensionality reduction using PCA • Implementation of SVM • Implementation of Decision tree on real word data set • Implementation of Naïve Bayes Classifier • Implementation of k-means clustering • Implementation of ANN	20

Text Book:

- Alpaydin, E., ". Introduction to machine learning", MIT press, 2009.
- Bishop, C. M. ,"Pattern recognition and machine learning" , (information science and statistics) springer-verlag new york. Inc. Secaucus, NJ, USA, 2006.

Reference Books:

- Harrington, P., " Machine learning in action. Shelter Island" , NY: Manning Publications Co., 2012.

Course Outcome: After completion of Lab, student will be able to:

- CO1: Implement the basic concepts of machine learning including bias-variance tradeoff.
- CO2: Analyze data using regression and re-sampling methods.
- CO3: Perform supervise learning for classification.
- CO4: Apply and perform dimensionality reduction.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO3
CO2	PO2,PO3/PSO1,PSO3
CO3	PO1,PO4/PSO3,PSO4
CO4	PO4/PSO1,PSO3

BCSC0002: OBJECT ORIENTED PROGRAMMING

Objective: *This course introduces the Object-Oriented programming paradigm to students. It also teaches a student how to think objectively and model a Java program for solving real-world problems.*

Prerequisites: Basic programming, control structures, functions, and a suitable language like Python or Java.

CREDITS:03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	**Object-Oriented Programming:** Features of Object-Oriented Programming, Introduction to Object-Oriented Java Programming. **g Java Technology & Environment:** Understanding the compilation process of the JVM, JVM vs JDK vs JRE, Key Features of Java, Structure of a simple Java program. **Working with Java Primitive Data Types:** Strongly Typed nature of Java, Primitive Data Types in Java, The new 'var' keyword, Scope of a variable. **Accepting User Input in Java Programs:** using the Scanner class, using command line arguments. **Programming Constructs:** Sequence, Selection, Iteration & Transfer Statements, For-Each Loop. **Working with Java Arrays**: Declaring and Initializing One-Dimensional and Two-Dimensional Arrays in Java, Introduction to java. util. Arrays class. **The String API:** String Data Type, commonly used methods from the String API, String Tokenizer, String Builder & String Buffer. **Creating and Using Methods:** Signature of a method, Types of Methods, Overloading methods in a class, Static and Non-Static Methods. **Describing and Using Objects & Classes:** Declare the structure of a Java class, declaring members of a class (fields and methods), declaring and using Java Objects, lifecycle of an Object (creation, assignment, dereferencing and garbage collection), Constructors of a class, Overloading Constructors, Constructor chaining using 'this' and 'super' keyword. **Using Java Packages:** create and import Java packages and static imports, abstracting program logic to packages, creating executable main class, running the executable class inside a package. **Applying Encapsulation**: Using access modifiers with/in a class, principles of encapsulation. **Programming Abstractly Through Interfaces:** create and implement Interfaces for programs, private and default methods in Interfaces, declaring Abstract Classes, Constructors in Abstract Classes. Marker Interface, Functional Interfaces, Lambda Expressions in Java.	20
II	**Reusing Implementations using Inheritance:** Declaring Subclasses and Super classes, extend Abstract Classes, implementing Interfaces, exploring polymorphic behavior by overriding methods, Object Types vs Reference Types, differentiate overloading, overriding and hiding. **Exception Handling:** Exception Hierarchy, Need of Exception Handling, Checked Exceptions, Unchecked Exceptions and Errors, Try-Catch Blocks, Finally, Throw & Throws Keywords, creating and handling Custom Exceptions. **Threads in Java:** Life Cycle of a Thread, creating threads using Runnable and Thread, 'sleep ()', Thread Priorities. **Using Wrapper Classes:** Wrapper Classes in Java, Boxing-Unboxing-Auto Boxing-Auto Unboxing. **Generics & Collections:** Creating Generic classes, Generic Methods, Diamond Notation, Wildcards, Type Erasure, Collection Hierarchy, Base Interfaces, Lists, Sets and Maps.**The Stream API:** Introduction to the Stream API, using lambda expressions in Streams.**Regular Expressions:** Pattern and Matcher Class. **JDBC:** JDBC Drivers, Connecting to a MySQL Database, Driver Manager, Connection Interface, Statement Interface, Result Set Interface, Prepared Statements.	20

TEXT BOOK

- Herbert Schildt, "The Complete Reference, Java Eleventh Edition", Oracle Press.2019.

Reference Book:

- Cay S Hosrtmann, "Core Java Volume I—Fundamentals, Eleventh Edition", Pearson,2018.
- Rogers Cadenhead, "Sams Teach Yourself Java in 21 Days (Covers Java 11/12), 8th Edition", Pearson,2020.

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

- CO1: Understand the basics of Object-Oriented Programming paradigm.
- CO2: Construct the logical flow of programs by using the sequence, selection, iterations and transfer statements.
- CO3: Apply the concepts of Object-Oriented Programming to model programs in Classes, Abstract Classes, Interfaces and Enums, and simplify program function by dissecting it into methods.
- CO4: Understand accessibility of members in a program unit and create packages to prevent namespace collisions.
- CO5: Predict run-time errors in a program by examining program functioning.
- CO6: Show the parallel processing capabilities of a program using a multithreading concept.
- CO7: Experiment with the predefined classes and interfaces defined in the Collections Framework.
- CO8: Develop a program using JDBC connectivity to demonstrate data persistence.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO1,PSO2
CO2	PO1,PO3/PSO1,PSO2
CO3	PO1,PO2/PSO1,PSO2
CO4	PO1/PSO2,PSO4
CO5	PO1,PO2,PO4/PSO4
CO6	PO1,PO2, PO3/PSO2
CO7	PO1,PO2,PO11/PSO2
CO8	PO1,PO2,PO3/PSO1,PSO2

BCSC0801: OBJECT ORIENTED PROGRAMMING LAB

Objective: *The objective of this course is that students will study and learn Object Oriented Modeling and programming.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I& II	Programs in Java and python based on the concepts of: • Classes, Constructors, Polymorphism and Keyword Static. Programs based on the concepts of: • Inheritance, Multithreading Using Thread Class & Interface Runnable, String Handling, Generic Classes. Programs based on the concepts of: • Handling Database Connectivity. • Implementation of Collection Framework. Programs based on the concepts of: • Database Connectivity. • Retrieving Data from Database. • Parameters Passing, Execute many Method. • Cursor Attributes. • Invoke Stored Procedures. • Invoke Stored Functions.	24

Reference Books:

- Naughton, Schildt, "The Complete Reference JAVA2", 9th Edition, Oracle Press.
- Bhave&Patekar, "Programming with Java", Pearson Education
- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

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

- CO1: Implement object oriented language features.
- CO2: Design GUIs and Graphical programming.
- CO3: Design object oriented solutions for small systems involving database and event handling concepts.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1
CO2	PO3,PO5/PSO2
CO3	PO3,PO5/PSO4

BCSG 0153: DIGITAL LOGIC DESIGN

Objective: This course enables the students To learn the fundamental concepts of digital logic design, to study methods of logic expression simplification, to understand the procedure for the analysis of combinational and sequential logic circuits, and to understand terms used in the designing of a memory and programmable logic devices.

Prerequisites for Digital Logic Design: basic mathematics, binary number systems, Boolean algebra, and fundamental programming or circuit concepts.

CREDITS:03 **L-P-T-J:3-0-0-0**

Module No.	Contents	Teaching Hours
I	Introduction to digital systems, number system conversion, signed number representation, floating-point number representation, weighted (8421), non-weighted binary codes (excess-3 and gray code), error detection/correction code, and Hamming code. Basic and universal logic gates, realization of Boolean expressions using logic diagram, minterms, maxterms, SoP and PoS forms, simplification of Boolean function using two variables, three variables, and four variables K-Map, conversion from SoP to PoS and vice-versa. **Design and analysis of combinational circuits:** half adder, half subtractor, full adder, full subtractor, 4-bit parallel binary adder-subtractor, binary multiplier, magnitude comparator, multiplexer, implementation of a Boolean function using a multiplexer, implementation of higher-order multiplexers using lower order multiplexers, demultiplexer.	**20**
II	Encoder, priority encoder, decoder, implementation of Boolean functions using a decoder. Introduction to sequential circuits, SR latch, SR flip-flop, JK flip-flop, D flip-flop, T flip-flop, PS-NS table, excitation table, characteristic equation of flip-flops. Analysis of clocked sequential circuits, Mealy and Moore state machines, state table, and state diagram. Shift register, SISO, SIPO, PISO, PIPO, and universal shift register. Binary counter, ripple MOD-N (up/down) and MOD<N binary counter, synchronous binary counter, ring counter, Johnson's counter. Introduction to memory, RAM, ROM, and implementation of Boolean functions using ROM. Programmable logic: PAL architecture, PLA architecture, System on Chip (SoC) design.	**20**

Text Book:

- 1. S. Salivahanan& S. Asivazhagan, "Digital Circuit & Design", IInd Edition.
- 2. M. Morris Mano and M. D. Ciletti, "Digital Design" 4th Edition, Pearson Education.

Reference Books:

- John F Wakerly, Digital Design, Fourth Edition, Pearson/PHI,2006
- John M Yarbrough, Digital Logic Applications and Design, Thomas Learning,2002
- Charles H Roth, Fundamentals of Logic Design, Thomson Learning , 2003
- Donald P Leach and Albert Paul Malvino, Digital Principals and Applications, 6th
- Edition, TMH,2003.
- William H Gothmann, Digital Electronocs, 2nd Edition, PHI, 1982

Course Outcomes (CO): Upon completion of this course students will be able to:

- **CO1**: Understand number system conversion, signed numbers, and floating-point number representation.
- **CO2**: Understand 8421 weighted, non-weighted binary codes including excess-3 and gray code, and Hamming code.

- **CO3:** Understand basic and universal logic gates, SoP and PoS forms, and simplification of Boolean functions using K-Map.
- **CO4:** Construct combinational circuits, including adders, subtractors, adder-subtractors, multipliers, comparators, encoders, decoders, multiplexers, and demultiplexers.
- **CO5:** Construct sequential circuits which include flip-flop, shift register, ripple counter, synchronous counters, Johnson counter, ring counter.
- **CO6:** Analyse clocked sequential circuits using state table and state diagram.
- **CO7:** Understand memory, programmable logic device including PAL and PLA, and SoC design.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1
CO2	PO1,PO2/PSO4
CO3	PO1,PO3,PO3,PO5/PSO4
CO4	PO1,PO3,PO4,PO5/PSO1
CO5	PO1,PO5,PO10,/PSO1

BCSG 0752: DIGITAL SYSTEM DESIGN LAB

Objective: *The main aim of the lab is to better understand the design of combinational and sequential circuits.*

CREDITS:01 **L-P-T-J:0-2-0-0**

Module No.	Content	Teaching Hours
I	1. Introduction to the lab and testing of logic gate IC's. 2. Realization of Half Adder and Half Subtractor using logic gates. 3. Realization of full-adder & full subtractor using logic gates. 4. Realization of a 4-bit binary decoder/ demultiplexer. 5. Realization of decimal to BCD encoder using IC 74147. 6. Realization of a 4x1 multiplexer. 7. Implementation of SR, JK, T, and D flip-flops using logic gates. 8. Realization and implementation of serial in parallel out and parallel in serial out shift register. 9. Realization and implementation of a 2-bit up/down binary synchronous counter. 10. Realization and implementation of a 4-bit binary ripple counter using JK flip-flop. 11. A project based on combinational and sequential circuit.	24

Text Book:

- 1. S. Salivahanan& S. Asivazhagan, "Digital Circuit & Design", IInd Edition.
- 2. M. Morris Mano and M. D. Ciletti, "Digital Design" 4th Edition, Pearson Education.

Reference Books:

- John F Wakerly, Digital Design, Fourth Edition, Pearson/PHI,2006
- John M Yarbrough, Digital Logic Applications and Design, Thomas Learning,2002
- Charles H Roth, Fundamentals of Logic Design, Thomson Learning , 2003
- Donald P Leach and Albert Paul Malvino, Digital Principals and Applications, 6th
- Edition, TMH,2003.
- William H Gothmann, Digital Electronocs, 2nd Edition, PHI, 1982

Course Outcomes (CO): Upon completion of this lab students will be able to:

- **CO1**: Implement combinational and sequential circuits using ICs.
- **CO2**:Understand the working of various building blocks present in an Arithmetic Logic Unit (ALU) in a computer system.
- **CO3:** Understand basic and universal logic gates, SoP and PoS forms, and simplification of Boolean functions using K-Map.
- **CO4:** Construct sequential circuits which include flip-flop, shift register, ripple counter, synchronous counters, Johnson counter, ring counter.
- **CO5:** Construct sequential circuits which include flip-flop, shift register, ripple counter, synchronous counters, Johnson counter, ring counter.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO5/PSO1
CO2	PO1,PO2/PSO4
CO3	PO1,PO3,PO4,/PSO4
CO4	PO1,PO3,PO4,/PSO1
CO5	PO1,PO3,PO4,/PSO1,POS4

SEMESTER -IV
SYLLABUS

BCSC1005: COMPUTER ORGANIZATION

Objective: *This course aims to introduce the concept of computer organization. In particular, it focuses on basic architectural hardware issues that affect the nature and performance of software.*

Prerequisites: Basic knowledge of digital logic, binary arithmetic, computer fundamentals, and elementary programming concepts.

CREDITS: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Basic Organization:** Basic organization of the computer and Block level description of the functional units, Number representation; 1's and 2's Complement, Integer Representation, Arithmetic Addition & Subtraction with overflow. fixed and floating-point number representation, IEEE standard floating point representation. Introduction to Combinational Circuit- half adder, full adder, binary adder/subtractor, carry look ahead adders. Multiplexer and Demultiplexer, Register, bus and memory transfer. **Central Processing Unit:** Addition and subtraction of signed numbers, Multiplication: Signed operand multiplication, Booths algorithm. Processor organization, general registers organization, stack organization, Three, Two, One & Zero address instruction. Addressing modes, Micro- operations (Arithmetic, Logical & Shift) and its applications.	20
II	**Multiprogramming and Multiprocessing:** Flynn's classification, Introduction to pipelined operation. Instruction types, formats, Instruction cycles. **Control Unit:** Execution of a complete instruction. Hardwired and micro programmed control unit. Unconditional and Conditional branching. Microinstruction with next address field, pre-fetching microinstructions, Concept of horizontal and vertical microprogramming. **Memory:** Basic concept of Memory and its hierarchy, RAM memories, 2D, 2 & 1/2D memory organization. ROM memories. Cache memories: concept and design issues, performance, address mapping and replacement. Virtual memory: concept and implementation. **Input/Output:** Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Buses, bus architecture, types of buses and bus arbitration. Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors. Standard communication interfaces.	20

Text Books:

- M. Mano, "Computer System Architecture", 3rd Edition, PHI,1996

Reference Books:

- D.W. Patterson, "Computer Organization and Design", 4th **Edition**, Elsevier Publication, 2008.
- William Stalling, "Computer Organization",8th Edition, PHI, 2011.
- V. CarlHamacher, Zaky, "Computer Organization",4th International Edition, TMH, 1996.
- John P Hays, "Computer Organization", 2nd Edition, TMH.
- Tannenbaum , "Structured Computer Organization", 5thEdition, PHI, 2005.
- P Pal Chaudhry, "Computer Organization & Design", 2ndEdition, PHI, 2002.

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

- CO1: Understand the basics of digital computer system.

- CO2: Demonstrate the principle of arithmetic operations on unsigned, signed integers and floating point numbers.
- CO3: Understand the concepts of Combinational and Sequential circuits and their applications.
- CO4: Understand the CPU architecture and organization.
- CO5: Explain the basic concepts of pipelining.
- CO6: Design the steps for the execution of the complete instruction for hardwired and micro-programmed control unit.
- CO7: Explain the function of memory hierarchy.
- CO8: Determine the interface of CPU with input/output devices and their modes of transfer.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO1
CO2	PO1,PO3/PSO1
CO3	PO2,PO3,PO5/PSO2
CO4	PO2,PO3,PO4/PSO1,PSO3
CO5	PO2,PO3,PO4/PSO2
CO6	PO1,PO2,PO3/PSO1,PSO3
CO7	PO2,PO3,PO5/PSO2,PSO3
CO8	PO3,PO4/PSO1

BCSC0804: COMPUTER ORGANIZATION LAB

Objective: *The aim of the lab is to better understand the design of sequential Circuits such as Flip-Flops, Registers, and Counters.*

CREDITS: 01 **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I & II	• Bread Board Implementation of Flip-Flops. • Experiments with clocked Flip-Flops. • Design of Counters. • Bread Board implementation of Counters &Shift Registers. • Implementation of Arithmetic Algorithms. • Bread Board implementation of Adder/Subtraction (Half, Full). • Bread Board implementation of Binary Adder. • Bread Board implementation of Seven Segment Display. • Small Project based on combinational and sequential circuit.	24

Reference Books:

- D.W. Patterson , "Computer Organization and Design", 4th**Edition**, Elsevier Publication, 2008.
- William Stalling , "Computer Organization",8th Edition, PHI, 2011.
- M. Mano , "Computer System Architecture", 3rd Edition, PHI.

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

- CO1: Implement the Combinational and Sequential Circuit.
- CO2: Demonstrate the working of counter and shift register.
- CO3: Demonstrate the working of ALU and seven segment displays.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2,PO3,PO5/PSO2
CO2	PO3,PO4/PSO2
CO3	PO3,PO5/PSO1,PSO2

BCSC0006: DATA STRUCTURES AND ALGORITHMS

Objective: *The objective of this course is that students will construct and application of various data structures and abstract data types including lists, stacks, queues, trees and graphs.*

Prerequisite: Basic knowledge of C programming, problem-solving techniques, and fundamental mathematical logic including recursion and array handling.

CREDITS: 04 **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Basic Terminology, Elementary Data Organization, Properties of an Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic Notations – Big-Oh; Operations on Data Structure, Abstract Data Types (ADT). **Linked Lists:** Implementation of Singly Linked Lists, Doubly Linked List, Circular Linked List, Operations on a Linked List - Insertion, Deletion, Traversal; Generalized Linked List, Polynomial Representation and Addition. **Stacks:** Primitive Stack Operations - Push & Pop, Array and Linked Implementation of Stack in C, Application of Stack: Prefix and Postfix Expressions, Evaluation of Postfix Expression, conversion of Infix to Postfix expression, Recursion, Principles of Recursion, Tail Recursion, Removal of Recursion, use of stack in Recursion, Tower of Hanoi Problem. **Queues:** Operations on Queue - Add, Delete operations, Implementation of Queue Using Array and Linked List, Circular Queues, Deque and Priority Queue. **Trees:** Basic Terminology, Array Representation and Dynamic Representation; Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Tree Traversal Algorithms - Inorder, Preorder and Postorder; Threaded Binary Trees, Traversing Threaded Binary Trees.	20
II	**Search Trees:** Binary Search Trees (BST), Insertion and Deletion in BST, AVL Trees, Introduction to M-Way Search Trees, B Trees. **Searching:** Sequential Search, Binary Search. **Sorting:** Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Two Way Merge Sort, and Heap Sort. **Graphs:** Terminology, Adjacency Matrices, Adjacency List, Graph Traversal - Depth First Search and Breadth First Search; Spanning Trees, Minimum Cost Spanning Trees – Prim's and Kruskal's Algorithm; Shortest Path Algorithm – Bellman-Ford and Dijkstra's Algorithm. **Hashing & Indexing:** Hash Function, Collision Resolution Strategies. Primary Indices, Secondary Indices, Indexing and Hashing Comparisons.	20

Text Book:

- Aaron M. Tanenbaum, YedidyahLangsam and Moshe J. Augenstein, "Data Structures Using C and C++", 2nd Edition, PHI, 2009.

Reference Books:

- Horowitz and Sahani, "Fundamentals of Data Structures", 3rd Edition, W H Freeman & Co, 2004- 05.
- Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, TMH, 2007.
- R. Kruse, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2004.
- Lipschutz Schaum's Outline Series, "Data Structures", 12th Reprint, TMH, 2010.
- G A V Pai, "Data Structures and Algorithms", TMH, 2009.

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

- CO1: Understand the basic concepts of the data structure and algorithms.
- CO2: Understand the complexity representation in terms of Big Oh, Theta and Omega notations.
- CO3: Apply the associated operations in linear data structure like stack, Queue and link list.
- CO4: Apply the associated operations in Binary Search Tree, AVL Tree and M- Way Search Tree.

- CO5: Understand the basic algorithms such as heap sort, graph traversal, quick sort, AVL trees, and hashing.
- CO6: Select the appropriate data structure to solve the problem.
- CO7: Apply the shortest path algorithm to solve real life problem.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO1,PSO2
CO2	PO1, PO2/PSO1,PSO2
CO3	PO1/PSO1
CO4	PO1,PO4/PSO1
CO5	PO1,PO4/PSO3
CO6	PO2/PSO4
CO7	PO2/PSO4

BCSC0805: DATA STRUCTURES & ALGORITHMS LAB

***Objective*:** *The objective of this course is that students will understand and implement simple data structures, able demonstrate different sorting and searching techniques. and will be familiar with graphs and their applications.*

CREDITS:01 **L-T-P-J:0-0-2-0**

Module	Content	Lab Hours
I &II	• Program to implement various operations in a singly linked list. • Program to implement insertion, deletion and traversal in a doubly linked List. • Program to implement polynomial addition using linked list. • Program to demonstrate the various operations on stack. • Program to convert an infix expression into postfix expression. • Program to evaluate a given postfix expression. • Program to implement Tower of Hanoi problem using Recursion. • Program to demonstrate the implementation of various operations on linear and circular queue. • Program to demonstrate the implementation of insertion and traversals on a binary search tree. • Program to implement Dijkstra's Algorithm to find the shortest path between source and destination. • Program to search a given element as entered by the user using sequential and binary search to search a given element as entered by the user. • Implementation of various sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.	24

Note: All Code must be done in Java as well as Python

Text Book:

- Aaron M. Tanenbaum, YedidyahLangsam and Moshe J. Augenstein, "Data Structures Using C and C++", 2nd Edition, PHI, 2009.

Reference Books:

- Horowitz and Sahani, "Fundamentals of Data Structures", 3rd Edition, W H Freeman & Co, 2004- 05.
- Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, TMH, 2007.
- R. Kruse, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2004.
- Lipschutz Schaum's Outline Series, "Data Structures", 12th Reprint, TMH, 2010.
- G A V Pai, "Data Structures and Algorithms", TMH, 2009.

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

- CO1: Demonstrate the associated operations in linear data structure like stack, Queue and link list.
- CO2: Demonstrate the associated operations in Binary Search Tree and Dijkstra's Algorithm.
- CO3: Implementation the sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO1
CO2	PO4/PSO1,PSO3
CO3	PO2/PSO3,PSO4

BCSC1010: DISCRETE MATHEMATICS

Objective: *The objective is to introduce students to language and methods of the area of Discrete Mathematics. The focus of the module is on basic mathematical concepts in discrete mathematics and on applications of discrete mathematics in computer science.*

Prerequisite: Basic understanding of high school-level mathematics, including algebra, logic, and set theory fundamentals.

CREDITS: 4 **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Sets, Relations and Functions:** Introduction to Set Theory, Venn diagrams, algebra of Sets, Inclusion-Exclusion Principle, Partitions, Relations, Properties and their types, Function and their types. Recurrence Relations and Generating Functions **Introduction to Counting Principle**: Permutation, Combination, Permutation with Repetition, Combination with Repetition, Pigeonhole Principle. **Posets & Lattices:** Definition, Properties of lattices – Bounded, Complemented, Modular and Complete lattice. **Natural Numbers:** Introduction, Mathematical Induction, Variants of Induction, Induction with Nonzero Base cases. Proof Methods, Proof by counter – example, Proof by contradiction.	20
II	**Propositional Logic** - Logical Connectives, Truth Tables, Normal Forms (Conjunctive and Disjunctive), Validity; **Predicate Logic** - Quantifiers, Inference Theory **Algebra**: Motivation of Algebraic Structures, Finite Groups, Subgroups and Group Homomorphism; Lagrange's Theorem; Commutative Rings and Elementary Properties; **Graph Theory:** Trees: Definition, Binary tree, Binary tree traversal, Binary search tree. Introduction to Graphs, , Operations on Graphs, Representation of graphs, Types: Planner, Directed, Complete, Bipartite Graph, Isomorphism, Euler Graph, Hamiltonian Graph, Connectivity.	20

Text Book:

- Kenneth H Rosen , "Discrete Mathematics and Its Applications", 7th edition, TMH,2012.

Reference Books:

- J.P. Tremblay , "Discrete Mathematical Structures with Applications to Computer Science", TMH, New Delhi,1997.
- V. Krishnamurthy , "Combinatorics: Theory and Applications", East-West Press, New Delhi,1986.
- Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics- An Applied Introduction", 5th Edition, Pearson Education,2004.
- C.L. Liu, "Elements of Discrete Mathematics", 2nd Edition, TMH,2000.

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

- CO1: Understand the notion of mathematical thinking and proofs to solve the problem.
- CO2: Apply the basics of discrete probability and number theory to solve the real world problem. CO3: Analyze basic discrete structures and algorithms using effectively algebraic techniques.
- CO4**:** Analyze mathematical concepts like sets, reasoning, relational algebra and graph theory to solve optimization problems.
- CO5**.** Analyze the validity of an argument using logical notation.
- CO6. Demonstrate the basic structures of proof techniques to write and evaluate the

validity of arguments.

- CO7. Understand the basic principles of sets, set equalities and operations in sets.
- CO8. Apply counting principles to determine probabilities.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1,PSO3
CO2	PO1,PO3/PSO4
CO3	PO2,PO3/PSO3
CO4	PO2,PO3/PSO3
CO5	PO1,PO2/ PSO3
CO6	PO1,PO3/PSO2,PSO3
CO7	PO1,PO2/PSO1
CO8	PO1,PO3/PSO1,PSO4

XXXX: FULL STACK USING FLASK AND DJANGO

Objective: *To equip students with the knowledge and skills to develop, integrate, and deploy full stack web applications using Flask and Django frameworks.*

Prerequisite: Basic proficiency in Python programming, HTML/CSS, and foundational knowledge of web technologies and client-server architecture.

CREDIT-03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	Introduction to Web Development, HTTP, MVC, REST Setup Flask, create first app, HTML, CSS, JavaScript Essentials, Build static web pages, Flask Routing, Templates (Jinja2), Forms, Create dynamic pages, Handling POST/GET Requests, Form Validation, Build Flask forms with validation, Flask-SQL Alchemy & Database Integration, CRUD operations with SQLite/PostgreSQL, User Authentication: Flask-Login, Flask-Bcrypt, Register/Login/Logout, Flask Blueprints and Modularization Refactor app into modules, REST API Development with Flask-RESTful, Create and test APIs, JSON Handling, Postman, Token Auth (JWT), Secure REST APIs, Deployment: GitHub, Docker, Heroku, Deploy Flask App to cloud, Mini Project Week, Full-featured Flask app with CRUD + Auth	21
II	Django Overview, MVT, URL Routing, Views, Create Django project, Templates, Static Files, Template Tags, Dynamic HTML rendering, Django Models, Admin Interface, Create models and use admin, Django Forms & Validation. Input and model forms, Django Authentication System, Login/Signup, password handling, Django REST Framework: Serializers, Views, Create APIs for models, Class-Based Views & ViewSets, Use generic views and routers, API Security: Permissions, JWT, Throttling, Secure your Django APIs, Frontend-Backend Integration (Axios/Fetch), Connect frontend with Django APIs, Advanced Topics: Pagination, Filtering, Signal, Optimize backend logic, Capstone Project Full Stack App Development (Frontend Django DB), Testing, Debugging, Deployment (Docker/Vercel), CI/CD & Dockerize Django, Final Project Presentation & Viva, Project Demo	21

Text Books:

- Miguel G., Flask Web Development: Developing Web Applications with Python, Publisher: O'Reilly Media, ISBN: 9781491991732
- William S. Vincent – Django for Beginners: Build websites with Python and Django, Publisher: WelcomeToCode, ISBN: 9781735467207
- William S. Vincent – Django for APIs: Build web APIs with Python and Django , Publisher: WelcomeToCode, ISBN: 9781735467221

Reference Books:

- Antonio Mele – Django 4 By Example: Build powerful and reliable Python web applications from scratch
- Publisher: Packt Publishing, ISBN: 9781801813051.
- Rick van Hattem – Mastering Python Web Development, Publisher: Packt Publishing,ISBN: 9781785886774.
- Julia Elman & Mark Lavin – Lightweight Django: Using REST, WebSockets, and Backbone.js, Publisher: O'Reilly Media, ISBN: 9781491945940

Outcome: After completion of Lab, the student will be able to:

- CO1: To provide a solid foundation in full-stack web development by introducing core web technologies, backend architectures (MVC/MVT), and development tools.
- CO2: To develop proficiency in building backend applications using Flask and Django frameworks.
- CO3: To enable the creation of RESTful APIs and their integration with frontend technologies.
- CO4: To implement database-driven web solutions using SQL and ORM tools like SQLAlchemy and Django ORM, emphasising CRUD operations and data modelling.
- CO5: To implement real-world deployment by covering containerization, version control, cloud deployment, and full-stack project development with testing and debugging.
- CO6:To cultivate teamwork and project management skills through capstone project development.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4
CO6	PO1/PSO2,PSO4

XXXX: FULL STACK USING FLASK AND DJANGO LAB

***Objective*:** *The objective of this course is that students will understand and implement full stack development with Django.*

CREDIT:01 **L-P-T-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Build a simple Flask application with two routes and render content using HTML templates. • Create a contact form using Flask-WTF, validate inputs, and display submitted data. • Develop a basic CRUD (Create, Read, Update, Delete) app using Flask and SQLAlchemy. • Implement a user authentication system using Flask-Login with hashed passwords. • Develop a simple API to manage tasks or products and test it using Postman. • Set up a Django project and create a simple web app with views and templates. • Create models, register them in the admin panel, and perform CRUD via Django admin. • Create a REST API using Django REST Framework (DRF) with serializers and viewsets. • Develop a mini full-stack project (e.g., blog, task manager) using Flask or Django with frontend, database, authentication, and deployment.	20

Text Books:

- Miguel G., Flask Web Development: Developing Web Applications with Python, Publisher: O'Reilly Media, ISBN: 9781491991732
- William S. Vincent – Django for Beginners: Build websites with Python and Django, Publisher: WelcomeToCode, ISBN: 9781735467207
- William S. Vincent – Django for APIs: Build web APIs with Python and Django , Publisher: WelcomeToCode, ISBN: 9781735467221

Reference Books:

- Antonio Mele – Django 4 By Example: Build powerful and reliable Python web applications from scratch. Publisher: Packt Publishing, ISBN: 9781801813051.
- Rick van Hattem – Mastering Python Web Development, Publisher: Packt Publishing,ISBN: 9781785886774.
- Julia Elman & Mark Lavin – Lightweight Django: Using REST, WebSockets, and Backbone.js, Publisher: O'Reilly Media, ISBN: 9781491945940

Outcome: After completion of Lab, the student will be able to:

- CO1: To provide a solid foundation in full-stack web development by introducing core web technologies, backend architectures (MVC/MVT), and development tools.
- CO2: To develop proficiency in building backend applications using Flask and Django frameworks.
- CO3: To enable the creation of RESTful APIs and their integration with frontend technologies.
- CO4: To implement database-driven web solutions using SQL and ORM tools like SQLAlchemy and Django ORM, emphasising CRUD operations and data modelling.
- CO5: To implement real-world deployment by covering containerization, version control, cloud deployment, and full-stack project development with testing and debugging.
- CO6:To cultivate teamwork and project management skills through capstone project development.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4
CO6	PO1/PSO2,PSO4

BCSE XXXX: IOT AND ROBOTICS

Objective: To equip students with foundational knowledge and practical skills in Internet of Things (IoT) and Robotics by integrating sensor technology, microcontrollers, communication protocols, and autonomous systems, enabling them to design, develop, and deploy industry-relevant smart solutions.

Prerequisites: Basic knowledge of programming (Python/C), Fundamentals of electronics and circuits, Basic understanding of networking concepts

CREDITS: 03 **L-T-P-J:3-0-0-0**

Module No.	Contents	Teaching Hours (Approx.)
I	**Introduction to Iot,** Properties of Iot device, Iot Ecosystem, Iot Decision Framework, Iot Solution, Architecture Models, Major Iot Boards in the Market. Explore Ardiuno UNO and ESP32, Raspberry Pi, set up ESP32 and Raspberry Pi, show how Raspberry Pi works using SSH Client and TeamViewer, understand Sensing actions, and understand Actuators and MEMS **Communication Protocols used in IoT:** Types of communication-wired: UART,I2C and SPI, wireless: BLE, RFID, NFC, WiFi and LORA, properties, comparison of these devices, Major wireless Long-range communication devices, Point2point and Mesh network **Sensors:** analog and digital – Types, Analog sensors: LM35, Ultrasonic, and Passive infrared(PIR), Digital Sensors: Accelerometer, Magnetometer, Proximity Sensor, Gyroscope Calibration, Actuators: Types, stepper motor, servo motors, dc motors	20
II	**Introduction to Robotics Fundamentals of Robotics :** Robot anatomy-Definition, law of robotics, History and Terminology of Robotics-Accuracy and repeatability of Robotics-Simple problems Robot Classification, Robot Specification, notation, kinematic representations and transformations, dynamics techniques Introduction, the three levels of robot programming, ROS, requirements of a robot programming language, problems peculiar to robot programming languages **Robotics fundamentals**; Robot Kinematics: Position Analysis, Dynamic Analysis and Forces, Robot Programming languages & systems: trajectory planning and control. DDD concept, Intelligent robots, -Specifications of Robot-Speed of Robot, Robot joints and links-Robot classifications Architecture of robotic systems-Robot Drive systems-Hydraulic, Pneumatic and Electric system	20

Text Books:

- "Internet of Things: A Hands-On-Approach" by Arshdeep Bahga and Vijay Madisetti
- "Learning ROS for Robotics Programming" by Aaron Martinez and Enrique Fernández
- "Robotics, Vision and Control" by Peter Corke

Reference Books:

- Arshdeep Bahga, Vijay Madisetti – *Internet of Things: A Hands-On Approach*
- John J. Craig – *Introduction to Robotics: Mechanics and Control*
- Rajkumar Buyya, Amir Vahid Dastjerdi – *Internet of Things: Principles and Paradigms*
- Aaron Martinez, Enrique Fernández – *Learning ROS for Robotics Programming*
- Simon Monk – *Programming Arduino: Getting Started with Sketches*
- Brian W. Evans – *Beginning Arduino Programming*
- Marco Schwartz – *Internet of Things with the Arduino Yun*
- Online Resources & Open-Source Documentation – Arduino, Raspberry Pi, ESP32, ThingSpeak, Blynk, Firebase

Focus: This Course focus on Employability under CO1, CO2, CO4, CO5, CO6

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

- CO1: To provide foundational knowledge in IoT and Robotics for real-world industrial applications.
- CO2: To understand sensor interfacing, communication protocols, and microcontroller integration.
- CO3: To introduce robotics hardware and control mechanisms.
- CO4: To enable students to develop and prototype simple IoT-robotic systems.
- CO5: To encourage application-based learning through a hands-on industrial project.
- CO6: Work with open-source platforms and tools to prototype end-to-end applications

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO2
CO3	PO2,PO3,PO6,PO11/PSO1, PSO2,PSO4
CO4	PO1,PO3/PSO2
CO5	PO1,PO5/PSO1
CO6	PO2,PO3/PSO2

BCSE XXXX: IOT AND ROBOTICS LAB

Objective: To equip students with foundational knowledge and practical skills in Internet of Things (IoT) and Robotics by integrating sensor technology, microcontrollers, communication protocols, and autonomous systems, enabling them to design, develop, and deploy industry-relevant smart solutions.

Prerequisites: Basic knowledge of programming (Python/C), Fundamentals of electronics and circuits, Basic understanding of networking concepts.

CREDITS: 01 **L-T-P-J:0-0-2-0**

Module No.	Contents	Teaching Hours (Approx.)
I	• Install Arduino IDE, perform interfacing of Arduino UNO with LED for LED Blink experiment using sketch program • Install Raspberry Pi OS, connect peripherals, access via SSH, and update system packages to understand the basic setup and configuration of Raspberry Pi. • To control Raspberry Pi remotely using secure methods by Configure SSH access, connect via terminal, install TeamViewer Host, test screen sharing. • Write Python code to read sensor values and display in real-time by To interface temperature and humidity sensors and display data. • To demonstrate short-range wireless communication by the use of Raspberry Pi to send/receive data via Bluetooth and Wi-Fi modules. • To develop a PIR-based motion detection alert system that detect motion and trigger an alert; optionally send notification via email. • To build a robot that can follow a predefined path via. Implement line following logic using sensors and control algorithms. • To simulate robot motion planning between two points by modelling trajectory using position-time graphs and simulate movement with constraints. • To apply basic noise filtering to sensor data via. Accelerometer and gyroscope (MPU6050), Python/Arduino. Collect raw data, apply moving average or Kalman filter, and visualize results. • To control a robot's movement using hand gestures. Map gestures to directions and control the robot wirelessly.	20

Tools: Arduino IDE, ESP-IDF, Node-RED, Mosquitto MQTT, Python (paho-mqtt, requests), ThingWorx (REST API/MQTT), TinkerCAD, Fritzing, OpenCV, Roboflow, Flask, Streamlit, GitHub.

Focus: This Course focus on Employability under CO1, CO2, CO4, CO5, CO6

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

- CO1: To provide foundational knowledge in IoT and Robotics for real-world industrial applications.
- CO2: To understand sensor interfacing, communication protocols, and microcontroller integration.
- CO3: To introduce robotics hardware and control mechanisms.
- CO4: To enable students to develop and prototype simple IoT-robotic systems.
- CO5: To encourage application-based learning through a hands-on industrial project.
- CO6: Work with open-source platforms and tools to prototype end-to-end applications

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO2

CO3	PO2,PO3,PO6,PO11/PSO1, PSO2,PSO4
CO4	PO1,PO3/PSO2
CO5	PO1,PO5/PSO1
CO6	PO2,PO3/PSO2