VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

CURRICULUM AND SYLLABI

(2021-2022)

B. Tech. Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

VISION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

Transforming life through excellence in education and research.

MISSION STATEMENT OF VELLORE INSTITUTE OF TECHNOLOGY

World class Education: Excellence in education, grounded in ethics and critical thinking, for improvement of life.

Cutting edge Research: An innovation ecosystem to extend knowledge and solve critical problems.

Impactful People: Happy, accountable, caring and effective workforce and students.

Rewarding Co-creations: Active collaboration with national & international industries & universities for productivity and economic development.

Service to Society: Service to the region and world through knowledge and compassion.

VISION STATEMENT OF THE SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

To be a world-renowned centre of education, research and service in computing and allied domains.

MISSION STATEMENT OF THE SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

- To offer computing education programs with the goal that the students become technically competent and develop lifelong learning skill.
- To undertake path-breaking research that creates new computing technologies and solutions for industry and society at large.
- To foster vibrant outreach programs for industry, research organizations, academia and society.

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

B.Tech. Computer Science and Engineering

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

1. Graduates will be engineering practitioners and leaders, who would help solve industry's technological problems.
2. Graduates will be engineering professionals, innovators or entrepreneurs engaged in technology development, technology deployment, or engineering system implementation in industry.
3. Graduates will function in their profession with social awareness and responsibility.
4. Graduates will interact with their peers in other disciplines in industry and society and contribute to the economic growth of the country.
5. Graduates will be successful in pursuing higher studies in engineering or management.
6. Graduates will pursue career paths in teaching or research.

PROGRAMME OUTCOMES (POs)

PO_01: Having an ability to apply mathematics and science in engineering applications.

PO_02: Having a clear understanding of the subject related concepts and of contemporary issues and apply them to identify, formulate and analyse complex engineering problems.

PO_03: Having an ability to design a component or a product applying all the relevant standards and with realistic constraints, including public health, safety, culture, society and environment

PO_04: Having an ability to design and conduct experiments, as well as to analyse and interpret data, and synthesis of information

PO_05: Having an ability to use techniques, skills, resources and modern engineering and IT tools necessary for engineering practice

PO_06: Having problem solving ability- to assess social issues (societal, health, safety, legal and cultural) and engineering problems

PO_07: Having adaptive thinking and adaptability in relation to environmental context and sustainable development

PO_08: Having a clear understanding of professional and ethical responsibility

PO_09: Having cross cultural competency exhibited by working as a member or in teams

PO_10: Having a good working knowledge of communicating in English – communication with engineering community and society

PO_11: Having a good cognitive load management skills related to project management and finance

PO_12: Having interest and recognise the need for independent and lifelong learning

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

B.Tech. Computer Science and Engineering

PROGRAMME SPECIFIC OUTCOMES (PSOs)

1. Apply computing theory, languages and algorithms, as well as mathematical and statistical models, and the principles of optimization to appropriately formulate and use data analysis.

2. Apply the principles and techniques of database design, administration, and implementation to enhance data collection capabilities and decision-support systems. Ability to critique the role of information and analytics in supporting business processes and functions.

3. Invent and use appropriate models of data analysis, assess the quality of input, derive insight from results, and investigate potential issues. Also to organize big data sets into meaningful structures, incorporating data profiling and quality standards.

School of Computer Science and Engineering

B.Tech. Computer Science and Engineering

CREDIT INFO		
S.no	**Catagory**	**Credit**
1	Foundation Core	55
2	Discipline-linked Engineering Sciences	12
3	Discipline Core	44
4	Discipline Elective	15
5	Projects and Internship	9
6	Open Elective	15
7	Bridge Course	0
8	Non-graded Core Requirement	11
Total Credits		161

Foundation Core									
sl.no	**Course Code**	**Course Title**	**Course Type**	**Version**	**L**	**T**	**P**	**J**	**Credit**
1	BCHY101L	Engineering Chemistry	Theory Only	1.0	3	0	0	0	3.0
2	BCHY101P	Engineering Chemistry Lab	Lab Only	1.0	0	0	2	0	1.0
3	BCSE101E	Computer Programming: Python	Embedded Theory and Lab	1.0	1	0	4	0	3.0
4	BCSE102L	Structured and Object-Oriented Programming	Theory Only	1.0	2	0	0	0	2.0
5	BCSE102P	Structured and Object-Oriented Programming Lab	Lab Only	1.0	0	0	4	0	2.0
6	BCSE103E	Computer Programming: Java	Embedded Theory and Lab	1.0	1	0	4	0	3.0
7	BECE101L	Basic Electronics	Theory Only	1.0	2	0	0	0	2.0
8	BECE101P	Basic Electronics Lab	Lab Only	1.0	0	0	2	0	1.0
9	BEEE101L	Basic Electrical Engineering	Theory Only	1.0	2	0	0	0	2.0
10	BEEE101P	Basic Electrical Engineering Lab	Lab Only	1.0	0	0	2	0	1.0
11	BENG101L	Technical English Communication	Theory Only	1.0	2	0	0	0	2.0
12	BENG101P	Technical English Communication Lab	Lab Only	1.0	0	0	2	0	1.0
13	BENG102P	Technical Report Writing	Lab Only	1.0	0	0	2	0	1.0
14	BFLE200L	B.Tech. Foreign Language - 2021	Basket	1.0	0	0	0	0	2.0
15	BHSM200L	B.Tech. HSM Elective - 2021	Basket	1.0	0	0	0	0	3.0
16	BMAT101L	Calculus	Theory Only	1.0	3	0	0	0	3.0
17	BMAT101P	Calculus Lab	Lab Only	1.0	0	0	2	0	1.0
18	BMAT102L	Differential Equations and Transforms	Theory Only	1.0	3	1	0	0	4.0
19	BMAT201L	Complex Variables and Linear Algebra	Theory Only	1.0	3	1	0	0	4.0
20	BMAT202L	Probability and Statistics	Theory Only	1.0	3	0	0	0	3.0
21	BMAT202P	Probability and Statistics Lab	Lab Only	1.0	0	0	2	0	1.0
22	BPHY101L	Engineering Physics	Theory Only	1.0	3	0	0	0	3.0
23	BPHY101P	Engineering Physics Lab	Lab Only	1.0	0	0	2	0	1.0
24	BSTS101P	Quantitative Skills Practice I	Soft Skill	1.0	0	0	3	0	1.5
25	BSTS102P	Quantitative Skills Practice II	Soft Skill	1.0	0	0	3	0	1.5
26	BSTS201P	Qualitative Skills Practice I	Soft Skill	1.0	0	0	3	0	1.5

Foundation Core									
27	BSTS202P	Qualitative Skills Practice II	Soft Skill	1.0	0	0	3	0	1.5

Discipline-linked Engineering Sciences									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	BECE102L	Digital Systems Design	Theory Only	1.0	3	0	0	0	3.0
2	BECE102P	Digital Systems Design Lab	Lab Only	1.0	0	0	2	0	1.0
3	BECE204L	Microprocessors and Microcontrollers	Theory Only	1.0	3	0	0	0	3.0
4	BECE204P	Microprocessors and Microcontrollers Lab	Lab Only	1.0	0	0	2	0	1.0
5	BMAT205L	Discrete Mathematics and Graph Theory	Theory Only	1.0	3	1	0	0	4.0

Discipline Core									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	BCSE202L	Data Structures and Algorithms	Theory Only	1.0	3	0	0	0	3.0
2	BCSE202P	Data Structures and Algorithms Lab	Lab Only	1.0	0	0	2	0	1.0
3	BCSE204L	Design and Analysis of Algorithms	Theory Only	1.0	3	0	0	0	3.0
4	BCSE204P	Design and Analysis of Algorithms Lab	Lab Only	1.0	0	0	2	0	1.0
5	BCSE205L	Computer Architecture and Organization	Theory Only	1.0	3	0	0	0	3.0
6	BCSE301L	Software Engineering	Theory Only	1.0	3	0	0	0	3.0
7	BCSE301P	Software Engineering Lab	Lab Only	1.0	0	0	2	0	1.0
8	BCSE302L	Database Systems	Theory Only	1.0	3	0	0	0	3.0
9	BCSE302P	Database Systems Lab	Lab Only	1.0	0	0	2	0	1.0
10	BCSE303L	Operating Systems	Theory Only	1.0	3	0	0	0	3.0
11	BCSE303P	Operating Systems Lab	Lab Only	1.0	0	0	2	0	1.0
12	BCSE304L	Theory of Computation	Theory Only	1.0	3	0	0	0	3.0
13	BCSE305L	Embedded Systems	Theory Only	1.0	3	0	0	0	3.0
14	BCSE306L	Artificial Intelligence	Theory Only	1.0	3	0	0	0	3.0
15	BCSE307L	Compiler Design	Theory Only	1.0	3	0	0	0	3.0
16	BCSE307P	Compiler Design Lab	Lab Only	1.0	0	0	2	0	1.0
17	BCSE308L	Computer Networks	Theory Only	1.0	3	0	0	0	3.0
18	BCSE308P	Computer Networks Lab	Lab Only	1.0	0	0	2	0	1.0
19	BCSE309L	Cryptography and Network Security	Theory Only	1.0	3	0	0	0	3.0
20	BCSE309P	Cryptography and Network Security Lab	Lab Only	1.0	0	0	2	0	1.0

Discipline Elective									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	BCSE206L	Foundations of Data Science	Theory Only	1.0	3	0	0	0	3.0

Discipline Elective									
2	BCSE207L	Programming for Data Science	Theory Only	1.0	2	0	0	0	2.0
3	BCSE207P	Programming for Data Science Lab	Lab Only	1.0	0	0	2	0	1.0
4	BCSE208L	Data Mining	Theory Only	1.0	2	0	0	0	2.0
5	BCSE208P	Data Mining Lab	Lab Only	1.0	0	0	2	0	1.0
6	BCSE209L	Machine Learning	Theory Only	1.0	3	0	0	0	3.0
7	BCSE209P	Machine Learning Lab	Lab Only	1.0	0	0	2	0	1.0
8	BCSE310L	IoT Architectures and Protocols	Theory Only	1.0	3	0	0	0	3.0
9	BCSE311L	Sensors and Actuator Devices	Theory Only	1.0	2	0	0	0	2.0
10	BCSE311P	Sensors and Actuator Devices Lab	Lab Only	1.0	0	0	2	0	1.0
11	BCSE312L	Programming for IoT Boards	Theory Only	1.0	2	0	0	0	2.0
12	BCSE312P	Programming for IoT Boards Lab	Lab Only	1.0	0	0	2	0	1.0
13	BCSE313L	Fundamentals of Fog and Edge Computing	Theory Only	1.0	3	0	0	0	3.0
14	BCSE314L	Privacy and Security in IoT	Theory Only	1.0	3	0	0	0	3.0
15	BCSE315L	Wearable Computing	Theory Only	1.0	3	0	0	0	3.0
16	BCSE316L	Design of Smart Cities	Theory Only	1.0	3	0	0	0	3.0
17	BCSE317L	Information Security	Theory Only	1.0	3	0	0	0	3.0
18	BCSE318L	Data Privacy	Theory Only	1.0	3	0	0	0	3.0
19	BCSE319L	Penetration Testing and Vulnerability Analysis	Theory Only	1.0	2	0	0	0	2.0
20	BCSE319P	Penetration Testing and Vulnerability Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
21	BCSE320L	Web Application Security	Theory Only	1.0	3	0	0	0	3.0
22	BCSE321L	Malware Analysis	Theory Only	1.0	2	0	0	0	2.0
23	BCSE321P	Malware Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
24	BCSE322L	Digital Forensics	Theory Only	1.0	2	0	0	0	2.0
25	BCSE322P	Digital Forensics Lab	Lab Only	1.0	0	0	2	0	1.0
26	BCSE323L	Digital Watermarking and Steganography	Theory Only	1.0	3	0	0	0	3.0
27	BCSE324L	Foundations of Blockchain Technology	Theory Only	1.0	3	0	0	0	3.0
28	BCSE325L	Introduction to Bitcoin	Theory Only	1.0	3	0	0	0	3.0
29	BCSE326L	Blockchain Architecture Design	Theory Only	1.0	3	0	0	0	3.0
30	BCSE327L	Smart Contracts	Theory Only	1.0	2	0	0	0	2.0
31	BCSE327P	Smart Contracts Lab	Lab Only	1.0	0	0	2	0	1.0
32	BCSE328L	Cryptocurrency Technologies	Theory Only	1.0	3	0	0	0	3.0
33	BCSE329L	Blockchain and Distributed Ledger Technology	Theory Only	1.0	2	0	0	0	2.0
34	BCSE329P	Blockchain and Distributed Ledger Technology Lab	Lab Only	1.0	0	0	2	0	1.0
35	BCSE330L	Public Key Infrastructure and Trust Management	Theory Only	1.0	3	0	0	0	3.0
36	BCSE331L	Exploratory Data Analysis	Theory Only	1.0	2	0	0	0	2.0
37	BCSE331P	Exploratory Data Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
38	BCSE332L	Deep Learning	Theory Only	1.0	3	0	0	0	3.0
39	BCSE332P	Deep Learning Lab	Lab Only	1.0	0	0	2	0	1.0
40	BCSE333L	Statistical Inference	Theory Only	1.0	2	0	0	0	2.0
41	BCSE333P	Statistical Inference Lab	Lab Only	1.0	0	0	2	0	1.0
42	BCSE334L	Predictive Analytics	Theory Only	1.0	3	0	0	0	3.0
43	BCSE335L	Healthcare Data Analytics	Theory Only	1.0	3	0	0	0	3.0
44	BCSE336L	Financial Data Analytics	Theory Only	1.0	2	0	0	0	2.0

Discipline Elective									
45	BCSE336P	Financial Data Analytics Lab	Lab Only	1.0	0	0	2	0	1.0
46	BCSE391J	Technical Answers to Real Problems Project	Project	1.0	0	0	0	0	3.0
47	BCSE392J	Design Project	Project	1.0	0	0	0	0	3.0
48	BCSE393J	Laboratory Project	Project	1.0	0	0	0	0	3.0
49	BCSE394J	Product Development Project	Project	1.0	0	0	0	0	3.0
50	BCSE396J	Reading Course	Project	1.0	0	0	0	0	3.0
51	BCSE397J	Special Project	Project	1.0	0	0	0	0	3.0
52	BCSE398J	Simulation Project	Project	1.0	0	0	0	0	3.0
53	BEEE303L	Control Systems	Theory Only	1.0	3	0	0	0	3.0
54	BEEE303P	Control Systems Lab	Lab Only	1.0	0	0	2	0	1.0

Projects and Internship									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	BCSE399J	Summer Industrial Internship	Project	1.0	0	0	0	0	1.0
2	BCSE497J	Project - I	Project	1.0	0	0	0	0	3.0
3	BCSE498J	Project - II / Internship	Project	1.0	0	0	0	0	5.0
4	BCSE499J	One Semester Internship	Project	1.0	0	0	0	0	14.0

Open Elective									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	CFOC102M	Introduction to Cognitive Psychology	Online Course	1.0	0	0	0	0	3.0
2	CFOC103M	Introduction to Political Theory	Online Course	1.0	0	0	0	0	3.0
3	CFOC104M	Six Sigma	Online Course	1.0	0	0	0	0	3.0
4	CFOC105M	Emotional Intelligence	Online Course	1.0	0	0	0	0	2.0
5	CFOC109M	Design Thinking - A Primer	Online Course	1.0	0	0	0	0	1.0
6	CFOC118M	Practical Machine Learning with Tensorflow	Online Course	1.0	0	0	0	0	2.0
7	CFOC122M	Educational Leadership	Online Course	1.0	0	0	0	0	2.0
8	CFOC133M	E-Business	Online Course	1.0	0	0	0	0	3.0
9	CFOC152M	Pattern Recognition and Application	Online Course	1.0	0	0	0	0	3.0
10	CFOC165M	Software testing	Online Course	1.0	0	0	0	0	3.0
11	CFOC188M	Ethical Hacking	Online Course	1.0	0	0	0	0	3.0
12	CFOC190M	Positive Psychology	Online Course	1.0	0	0	0	0	2.0
13	CFOC191M	Forests and their Management	Online Course	1.0	0	0	0	0	3.0
14	CFOC193M	Bioengineering: An Interface with Biology and Medicine	Online Course	1.0	0	0	0	0	2.0
15	CFOC197M	Bio-Informatics: Algorithms and Applications	Online Course	1.0	0	0	0	0	3.0
16	CFOC203M	Natural Hazards	Online Course	1.0	0	0	0	0	2.0
17	CFOC207M	Electronic Waste Management - Issues And Challenges	Online Course	1.0	0	0	0	0	1.0

Open Elective									
18	CFOC227M	GPU Architectures and Programming	Online Course	1.0	0	0	0	0	3.0
19	CFOC232M	Consumer Behaviour	Online Course	1.0	0	0	0	0	2.0
20	CFOC235M	Rocket Propulsion	Online Course	1.0	0	0	0	0	3.0
21	CFOC236M	Aircraft Maintenance	Online Course	1.0	0	0	0	0	1.0
22	CFOC253M	Plastic Waste Management	Online Course	1.0	0	0	0	0	2.0
23	CFOC258M	Introduction to Geographic Information Systems	Online Course	1.0	0	0	0	0	1.0
24	CFOC282M	Waste to Energy Conversion	Online Course	1.0	0	0	0	0	2.0
25	CFOC329M	Design, Technology and Innovation	Online Course	1.0	0	0	0	0	2.0
26	CFOC332M	Fundamentals of Automotive Systems	Online Course	1.0	0	0	0	0	3.0
27	CFOC356M	Analog Circuits	Online Course	1.0	0	0	0	0	3.0
28	CFOC365M	Evolution of Air Interface towards 5G	Online Course	1.0	0	0	0	0	2.0
29	CFOC384M	Entrepreneurship Essentials	Online Course	1.0	0	0	0	0	3.0
30	CFOC388M	Energy Resources, Economics and Environment	Online Course	1.0	0	0	0	0	3.0
31	CFOC391M	Effective Writing	Online Course	1.0	0	0	0	0	1.0
32	CFOC395M	Speaking Effectively	Online Course	1.0	0	0	0	0	2.0
33	CFOC397M	Intellectual Property	Online Course	1.0	0	0	0	0	3.0
34	CFOC400M	Language and Mind	Online Course	1.0	0	0	0	0	2.0
35	CFOC401M	The Nineteenth - Century English Novel	Online Course	1.0	0	0	0	0	3.0
36	CFOC402M	Introduction to World Literature	Online Course	1.0	0	0	0	0	3.0
37	CFOC405M	Economic Growth & Development	Online Course	1.0	0	0	0	0	2.0
38	CFOC407M	Introduction to Modern Indian Political Thought	Online Course	1.0	0	0	0	0	3.0
39	CFOC408M	English Literature of the Romantic Period, 1798 - 1832	Online Course	1.0	0	0	0	0	2.0
40	CFOC416M	Feminism : Concepts and Theories	Online Course	1.0	0	0	0	0	3.0
41	CFOC419M	Basic Real Analysis	Online Course	1.0	0	0	0	0	3.0
42	CFOC442M	Robotics and Control : Theory and Practice	Online Course	1.0	0	0	0	0	2.0
43	CFOC475M	IC Engines and Gas Turbines	Online Course	1.0	0	0	0	0	3.0
44	CFOC488M	Business Analytics For Management Decision	Online Course	1.0	0	0	0	0	3.0
45	CFOC490M	Sales and Distribution Management	Online Course	1.0	0	0	0	0	2.0
46	CFOC493M	Management of Inventory Systems	Online Course	1.0	0	0	0	0	3.0
47	CFOC494M	Quality Design And Control	Online Course	1.0	0	0	0	0	3.0
48	CFOC495M	Foundation Course in Managerial Economics	Online Course	1.0	0	0	0	0	2.0
49	CFOC496M	Engineering Econometrics	Online Course	1.0	0	0	0	0	3.0
50	CFOC497M	Financial Statement Analysis and Reporting	Online Course	1.0	0	0	0	0	3.0
51	CFOC498M	Business Statistics	Online Course	1.0	0	0	0	0	3.0
52	CFOC499M	Global Marketing Management	Online Course	1.0	0	0	0	0	2.0
53	CFOC500M	Marketing Research and Analysis - II	Online Course	1.0	0	0	0	0	3.0
54	CFOC503M	Marketing Analytics	Online Course	1.0	0	0	0	0	3.0
55	CFOC505M	Management of Commercial Banking	Online Course	1.0	0	0	0	0	3.0
56	CFOC508M	Entrepreneurship	Online Course	1.0	0	0	0	0	3.0
57	CFOC550M	Numerical Analysis	Online Course	1.0	0	0	0	0	4.0
58	CFOC570M	Public Speaking	Online Course	1.0	0	0	0	0	3.0
59	CFOC571M	Introduction To CFD	Online Course	1.0	0	0	0	0	3.0

Open Elective									
60	CFOC573M	Fundamentals Of Food Process Engineering	Online Course	1.0	0	0	0	0	3.0
61	CFOC575M	Wildlife Ecology	Online Course	1.0	0	0	0	0	3.0
62	CFOC576M	Integrated Waste Management For A Smart City	Online Course	1.0	0	0	0	0	3.0
63	CFOC577M	Introduction To Multimodal Urban Transportation Systems (MUTS)	Online Course	1.0	0	0	0	0	3.0
64	CFOC578M	Wastewater Treatment And Recycling	Online Course	1.0	0	0	0	0	3.0
65	CFOC580M	Real-Time Systems	Online Course	1.0	0	0	0	0	3.0
66	CFOC581M	Algorithmic Game Theory	Online Course	1.0	0	0	0	0	3.0
67	CFOC582M	Computational Number Theory and Algebra	Online Course	1.0	0	0	0	0	3.0
68	CFOC583M	Power System Protection	Online Course	1.0	0	0	0	0	3.0
69	CFOC584M	Accreditation And Outcome Based Learning	Online Course	1.0	0	0	0	0	2.0
70	CFOC587M	Economics of Banking and Finance Markets	Online Course	1.0	0	0	0	0	3.0
71	CFOC588M	Concepts Of Thermodynamics	Online Course	1.0	0	0	0	0	3.0
72	CFOC589M	Engineering Drawing And Computer Graphics	Online Course	1.0	0	0	0	0	3.0
73	CFOC591M	Principles Of Management	Online Course	1.0	0	0	0	0	3.0
74	CFOC592M	Stress Management	Online Course	1.0	0	0	0	0	1.0
75	CFOC593M	Corporate Finance	Online Course	1.0	0	0	0	0	3.0
76	CFOC594M	Customer Relationship Management	Online Course	1.0	0	0	0	0	2.0
77	CFOC597M	Globalization And Culture	Online Course	1.0	0	0	0	0	2.0
78	CFOC598M	Elements of Visual Representation	Online Course	1.0	0	0	0	0	2.0

Bridge Course									
sl.no	**Course Code**	**Course Title**	**Course Type**	**Version**	**L**	**T**	**P**	**J**	**Credit**
1	BENG101N	Effective English Communication	Lab Only	1.0	0	0	4	0	2.0

Non-graded Core Requirement									
sl.no	**Course Code**	**Course Title**	**Course Type**	**Version**	**L**	**T**	**P**	**J**	**Credit**
1	BCHY102N	Environmental Sciences	Online Course	1.0	0	0	0	0	2.0
2	BCSE101N	Introduction to Engineering	Project	1.0	0	0	0	0	1.0
3	BEXC100N	Extracurricular Activities / Co-Curricular Activities - B.Tech. Programmes	Basket	1.0	0	0	0	0	2.0
4	BHUM101N	Ethics and Values	Online Course	1.0	0	0	0	0	2.0
5	BSSC101N	Essence of Traditional Knowledge	Online Course	1.0	0	0	0	0	2.0
6	BSSC102N	Indian Constitution	Online Course	1.0	0	0	0	0	2.0

Fountation Core

BCSE101E	Computer Programming: Python	L	T	P	C
		1	0	4	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To provide exposure to basic problem-solving techniques using computers.
2. To inculcate the art of logical thinking abilities and propose novel solutions for real world problems through programming language constructs.

Course Outcome

1. Classify various algorithmic approaches, categorize the appropriate data representation, and demonstrate various control constructs.
2. Choose appropriate programming paradigms, interpret and handle data using files to propose solution through reusable modules; idealize the importance of modules and packages.

Module:1	**Introduction to Problem Solving**	**1 hour**
Problem Solving: Definition and Steps, Problem Analysis Chart, Developing an Algorithm, Flowchart and Pseudocode.		
Module:2	**Python Programming Fundamentals**	**2 hours**
Introduction to python – Interactive and Script Mode – Indentation – Comments – Variables – Reserved Words – Data Types – Operators and their precedence – Expressions – Built-in Functions – Importing from Packages.		
Module:3	**Control Structures**	**2 hours**
Decision Making and Branching: if, if-else, nested if, multi-way if-elif statements – Looping: while loop, for loop – else clauses in loops, nested loops – break, continue and pass statements.		
Module:4	**Collections**	**3 hours**
Lists: Create, Access, Slicing, Negative indices, List methods, List comprehensions – Tuples: Create, Indexing and slicing, Operations on tuples – Dictionary: Create, add, and replace values, Operations on dictionaries – Sets: Creation and operations.		
Module:5	**Strings and Regular Expressions**	**2 hours**
Strings: Comparison, Formatting, Slicing, Splitting, Stripping – Regular Expressions: Matching, Search and replace, Patterns.		
Module:6	**Functions and Files**	**3 hours**
Functions – Parameters and Arguments: Positional arguments, Keyword arguments, Parameters with default values – Local and Global scope of variables – Functions with Arbitrary arguments – Recursive Functions – Lambda Function. Files: Create, Open, Read, Write, Append and Close – tell and seek methods.		
Module:7	**Modules and Packages**	**2 hours**
Built-in modules – User-Defined modules – Overview of Numpy and Pandas packages.		
	Total Lecture hours:	**15 hours**

Text Book(s)

1.	Eric Matthes, Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2nd Edition, No starch Press, 2019

Reference Books

1.	Martic C Brown, Python: The Complete Reference, 4th Edition, McGraw Hill Publishers, 2018.
2.	John V. Guttag, Introduction to computation and programming using python: with applications to understanding data. 2nd Edition, MIT Press, 2016.

Mode of Evaluation: No separate evaluation for theory component.			
Indicative Experiments			
1.	Problem Analysis Chart, Flowchart and Pseudocode Practices.		
2.	Sequential Constructs using Python Operators, Expressions.		
3.	Branching (if, if-else, nested if, multi-way if-elif statements) and Looping (for, while, nested looping, break, continue, else in loops).		
4.	List, Tuples, Dictionaries & Sets.		
5.	Strings, Regular Expressions.		
6.	Functions, Lambda, Recursive Functions and Files.		
7.	Modules and Packages (NumPy and Pandas)		
Total Laboratory Hours			**60 hours**
Text Book(s)			
1.	Mariano Anaya, Clean Code in Python: Develop maintainable and efficient code, 2nd Edition, Packt Publishing Limited, 2021.		
Reference Books			
1.	Harsh Bhasin, Python for beginners, 1st Edition, New Age International (P) Ltd., 2019,		
	Mode of assessment: Continuous assessments and FAT		
Recommended by Board of Studies	03.07.2021		
Approved by Academic Council	No. 63	Date	23.09.2021

BCSE102L	Structured and Object-Oriented Programming	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart the basic constructs in structured programming and object-oriented programming paradigms.
2. To inculcate the insights and benefits in accessing memory locations by implementing real world problems.
3. To help solving real world problems through appropriate programming paradigms.

Course Outcome

At the end of the course, students should be able to:

1. Understand different programming language constructs and decision-making statements; manipulate data as a group.
2. Recognize the application of modular programming approach; create user defined data types and idealize the role of pointers.
3. Comprehend various elements of object-oriented programing paradigm; propose solutions through inheritance and polymorphism; identify the appropriate data structure for the given problem and devise solution using generic programming techniques.

Module:1	**C Programming Fundamentals**	**2 hours**

Variables - Reserved words – Data Types – Operators – Operator Precedence - Expressions - Type Conversions - I/O statements - Branching and Looping: if, if-else, nested if, if-else ladder, switch statement, goto statement - Loops: for, while and do…while – break and continue statements.

Module:2	**Arrays and Functions**	**4 hours**

Arrays: One Dimensional array - Two-Dimensional Array – Strings and its operations. User Defined Functions: Declaration – Definition – call by value and call by reference - Types of Functions - Recursive functions - Storage Classes - Scope, Visibility and Lifetime of Variables.

Module:3	**Pointers**	**4 hours**

Declaration and Access of Pointer Variables, Pointer arithmetic – Dynamic memory allocation – Pointers and arrays - Pointers and functions.

Module:4	**Structure and Union**	**2 hours**

Declaration, Initialization, Access of Structure Variables - Arrays of Structure - Arrays within Structure - Structure within Structures - Structures and Functions – Pointers to Structure -

Module:5	**Overview of Object-Oriented Programming**	**5 hours**

Features of OOP - Classes and Objects - “this” pointer - Constructors and Destructors - Static Data Members, Static Member Functions and Objects - Inline Functions – Call by reference - Functions with default Arguments - Functions with Objects as Arguments - Friend Functions and Friend Classes.

Module:6	**Inheritance**	**5 hours**

Inheritance - Types of Inheritance: Single inheritance, Multiple Inheritance, Multi-level

Inheritance, Hierarchical Inheritance - Multipath Inheritance - Inheritance and constructors.		
Module:7	**Polymorphism**	**4 hours**
Function Overloading - Operator Overloading – Dynamic Polymorphism - Virtual Functions - Pure virtual Functions - Abstract Classes.		
Module:8	**Generic Programming**	**4 hours**
Function templates and class templates, Standard Template Library.		
	Total Lecture hours:	**30 hours**

Text Book(s)	
1.	Herbert Schildt, C: The Complete Reference, 4th Edition, McGraw Hill Education, 2017
2.	Herbert Schildt, C++: The Complete Reference, 4th Edition, McGraw Hill Education, 2017.
Reference Books	
1.	Yashavant Kanetkar, Let Us C: 17th Edition, BPB Publicaitons, 2020.
2.	Stanley Lippman and Josee Lajoie, C++ Primer, 5th Edition, Addison-Wesley publishers, 2012.
Mode of Evaluation: CAT / Written Assignment / Quiz / FAT / Project.	

Recommended by Board of Studies	03.07.2021		
Approved by Academic Council	No. 63	Date	23.09.2021

BCSE102P	Structured and Object-Oriented Programming Lab	L	T	P	C
		0	0	4	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart the basic constructs in structured programming and object-oriented programming paradigms.
2. To inculcate the insights and benefits in accessing memory locations by implementing real world problems.
3. To solve real world problems through appropriate programming paradigms.

Course Outcome

At the end of the course, students should be able to:

1. Understand different programming language constructs and decision-making statements; manipulate data as a group.
2. Recognize the application of modular programming approach; create user defined data types and idealize the role of pointers.
3. Comprehend various elements of object-oriented programing paradigm; propose solutions through inheritance and polymorphism; identify the appropriate data structure for the given problem and devise solution using generic programming techniques.

Indicative Experiments

1.	Programs using basic control structures, branching and looping
2.	Experiment the use of 1-D, 2-D arrays and strings and Functions
3.	Demonstrate the application of pointers
4.	Experiment structures and unions
5.	Programs on basic Object-Oriented Programming constructs.
6.	Demonstrate various categories of inheritance
7.	Program to apply kinds of polymorphism.
8.	Develop generic templates and Standard Template Libraries.
	Total Laboratory Hours: 60 hours

Text Book(s)

1. Robert C. Seacord, Effective C: An Introduction to Professional C Programming, 1st Edition, No Starch Press, 2020.

Reference Book(s)

1. Vardan Grigoryan and Shunguang Wu, Expert C++: Become a proficient programmer by learning coding best practices with C++17 and C++20's latest features, 1st Edition, Packt Publishing Limited, 2020.

Mode of assessment: Continuous assessments and FAT.

Recommended by Board of Studies	03.07.2021		
Approved by Academic Council	No. 63	Date	23.09.2021

BCSE103E	Computer Programming : Java	L	T	P	C
		1	0	4	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. To introduce the core language features of Java and understand the fundamentals of Object -Oriented programming in Java.
2. To develop the ability of using Java to solve real world problems.

Course Outcome:

At the end of this course, students should be able to:

1. Understand basic programming constructs; realize the fundamentals of Object Orientated Programming in Java; apply inheritance and interface concepts for enhancing code reusability.
2. Realize the exception handling mechanism; process data within files and use the data structures in the collection framework for solving real world problems.

Module:1	**Java Basics**	**2 hours**

OOP Paradigm - Features of Java Language - JVM - Bytecode - Java program structure – Basic programming constructs - data types - variables – Java naming conventions – operators.

Module:2	**Looping Constructs and Arrays**	**2 hours**

Control and looping constructs - Arrays – one dimensional and multi-dimensional – enhanced for loop – Strings - Wrapper classes.

Module:3	**Classes and Objects**	**2 hours**

Class Fundamentals – Access and non-access specifiers - Declaring objects and assigning object reference variables – array of objects – constructors and destructors – usage of "this" and "static" keywords.

Module:4	**Inheritance and Polymorphism**	**3 hours**

Inheritance – types –- use of "super" – final keyword - Polymorphism – Overloading and Overriding - abstract class – Interfaces.

Module:5	**Packages and Exception Handling**	**2 hours**

Packages: Creating and Accessing - Sub packages.
Exception Handling - Types of Exception - Control Flow in Exceptions - Use of try, catch, finally, throw, throws in Exception Handling - User defined exceptions.

Module:6	**IO Streams and Files**	**2 hours**

Java I/O streams – FileInputStream & FileOutputStream – FileReader & FileWriter- DataInputStream & DataOutputStream – BufferedInputStream & BufferedOutputStream – PrintOutputStream - Serialization and Deserialization.

Module:7	**Collection Framework**	**2 hours**

Generic classes and methods - Collection framework: List and Map.

	Total Lecture hours:	**15 hours**

Text Book(s)

1.	Y. Daniel Liang, "Introduction to Java programming" - comprehensive version-11th Edition, Pearson publisher, 2017.

Reference Books

1.	Herbert Schildt , The Complete Reference -Java, Tata McGraw-Hill publisher, 10th Edition, 2017.
2	Cay Horstmann,"Big Java", 4th edition, John Wiley & Sons publisher, 5th edition, 2015
3	E.Balagurusamy, "Programming with Java", Tata McGraw-Hill publishers, 6th edition, 2019

Mode of Evaluation: No separate evaluation for theory component.			
Indicative Experiments			
1.	Programs using sequential and branching structures.		
2.	Experiment the use of looping, arrays and strings.		
3.	Demonstrate basic Object-Oriented programming elements.		
4.	Experiment the use of inheritance, polymorphism and abstract classes.		
5.	Designing packages and demonstrate exception handling.		
6.	Demonstrate the use of IO streams, file handling and serialization.		
7.	Program to discover application of collections.		
	Total Laboratory Hours	**60 hours**	
Text Book(s)			
1.	Marc Loy, Patrick Niemeyer and Daniel Leuck, Learning Java, O'Reilly Media, Inc., 5th Edition, 2020.		
Reference Books			
1.	Dhruti Shah, 100+ Solutions in Java: A Hands-On Introduction to Programming in Java, BPB Publications, 1st Edition, 2020.		
Mode of assessment: Continuous assessments and FAT			
Recommended by Board of Studies		03.07.2021	
Approved by Academic Council	No. 63	Date	23.09.2021

Discipline-linked Engineering Sciences

Course Code	Course Title	L	T	P	C
BECE102L	Digital Systems Design	3	0	0	3
Pre-requisite	Nil	Syllabus version			
		1.0			

Course Objectives

1. Provide an understanding of Boolean algebra and logic functions.
2. Develop the knowledge of combinational and sequential logic circuit design.
3. Design and model the data path circuits for digital systems.
4. Establish a strong understanding of programmable logic.
5. Enable the student to design and model the logic circuits using Verilog HDL.

Course Outcome

At the end of the course the student will be able to

1. Optimize the logic functions using and Boolean principles and K-map.
2. Model the Combinational and Sequential logic circuits using Verilog HDL.
3. Design the various combinational logic circuits and data path circuits.
4. Analyze and apply the design aspects of sequential logic circuits.
5. Analyze and apply the design aspects of Finite state machines.
6. Examine the basic architectures of programmable logic devices.

Module:1	Digital Logic	8 hours

Boolean Algebra: Basic definitions, Axiomatic definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Simplification of Boolean functions. Gate-Level Minimization: The Map Method (K-map up to 4 variable), Product of Sums and Sum of Products Simplification, NAND and NOR Implementation. Logic Families: Digital Logic Gates, TTL and CMOS logic families.

Module:2	Verilog HDL	5 hours

Lexical Conventions, Ports and Modules, Operators, Dataflow Modelling, Gate Level Modelling, Behavioural Modeling, Test Bench.

Module:3	Design of Combinational Logic Circuits	8 hours

Design Procedure, Half Adder, Full Adder, Half Subtractor, Full Subtractor, Decoders, Encoders, Multiplexers, De-multiplexers, Parity generator and checker, Applications of Decoder, Multiplexer and De-multiplexer. Modeling of Combinational logic circuits using Verilog HDL.

Module:4	Design of data path circuits	6 hours

N-bit Parallel Adder/Subtractor, Carry Look Ahead Adder, Unsigned Array Multiplier, Booth Multiplier, 4-Bit Magnitude comparator. Modeling of data path circuits using Verilog HDL.

Module:5	Design of Sequential Logic Circuits	8 hours

Latches, Flip-Flops - SR, D, JK & T, Buffer Registers, Shift Registers - SISO, SIPO, PISO, PIPO, Design of synchronous sequential circuits: state table and state diagrams, Design of counters: Modulo-n, Johnson, Ring, Up/Down, Asynchronous counter. Modeling of sequential logic circuits using Verilog HDL.

Module:6	Design of FSM	4 hours

Finite state Machine(FSM):Mealy FSM and Moore FSM , Design Example : Sequence detection, Modeling of FSM using Verilog HDL.

Module:7	Programmable Logic Devices	4 hours

Types of Programmable Logic Devices: PLA, PAL, CPLD, FPGA Generic Architecture.

Module:8	**Contemporary issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Textbook(s)	
1.	M. Morris Mano and Michael D. Ciletti, Digital Design: With an Introduction to the Verilog HDL and System Verilog, 2018, 6th Edition, Pearson Pvt. Ltd.
Reference Books	
1.	Ming-Bo Lin, Digital Systems Design and Practice: Using Verilog HDL and FPGAs, 2015, 2nd Edition, Create Space Independent Publishing Platform.
2.	Samir Palnitkar, Verilog HDL: A Guide to Digital Design and Synthesis, 2009, 2nd edition, Prentice Hall of India Pvt. Ltd.
3.	Stephen Brown and ZvonkoVranesic, Fundamentals of Digital Logic with Verilog Design, 2013, 3rd Edition, McGraw-Hill Higher Education.

Mode of Evaluation: Continuous Assessment Test, Digital Assignment, Quiz and Final Assessment Test			
Recommended by Board of Studies	14-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course Code	Course Title	L	T	P	C
BECE102P	Digital Systems Design Lab	0	0	2	1
Pre-requisite	Nil	Syllabus version			
		1.0			

Course Objective

- To apply theoretical knowledge gained in the theory course and get hands-on experience of the topics.

Course Outcome

At the end of the course the student will be able to

1. Design, simulate and synthesize combinational logic circuits, data path circuits and sequential logic circuits using Verilog HDL.
2. Design and implement FSM on FPGA.
3. Design and implement small digital systems on FPGA.

Indicative Experiments

1.	Characteristics of Digital ICs, Realization of Boolean expressions	**2 hours**
2.	Design and Verilog modeling of Combinational Logic circuits	**4 hours**
3.	Design and Verilog modeling of various data path elements - Adders	**2 hours**
4.	Design and Verilog modeling of various data path elements - Multipliers	**2 hours**
5.	Implementation of combinational circuits – (FPGA / Trainer Kit)	**2 hours**
6.	Implementation of data path circuit - (FPGA / Trainer Kit)	**2 hours**
7.	Design and Verilog modeling of simple sequential circuits like Counters and Shift registers	**2 hours**
8.	Design and Verilog modeling of complex sequential circuits	**2 hours**
9.	Implementation of Sequential circuits - (FPGA / Trainer Kit)	**2 hours**
10.	Design and Verilog modeling of FSM based design – Serial Adder	**2 hours**
11.	Design and Verilog modeling of FSM based design – Traffic Light Controller / Vending Machine	**4 hours**
12.	Design of ALU	**4 hours**
	Total Laboratory Hours	**30 hours**

Mode of Assessment: Continuous Assessment and Final Assessment Test

Recommended by Board of Studies	14-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course Code	Course Title	L	T	P	C
BECE204L	Microprocessors and Microcontrollers	3	0	0	3
Pre-requisite	BECE102L	Syllabus version			
		1.0			

Course Objectives:

1. To acquaint students with architectures of Intel microprocessors, microcontroller and ARM processors.
2. To familiarize the students with assembly language programming in 8051 microcontroller and ARM processor.
3. To interface peripherals and I/O devices with the 8051 microcontroller.

Course Outcome:

At the end of the course, the student should be able to

1. Comprehend the various microprocessors including Intel Pentium Processors
2. Infer the architecture and Programming of Intel 8086 Microprocessor.
3. Comprehend the architectures and programming of 8051 microcontroller.
4. Deploy the implementation of various peripherals such as general purpose input/ output, timers, serial communication, LCD, keypad and ADC with 8051 microcontroller
5. Infer the architecture of ARM Processor
6. Develop the simple application using ARM processor.

Module:1	Overview of Microprocessors	3 hours

Introduction to Microprocessors, 8-bit/16-bit Microprocessor, Overview of Intel Pentium, I (i3, i5, i7) Series Processor.

Module:2	Microprocessor Architecture and Interfacing: Intel x86	8 hours

16-bit Microprocessor: 8086 - Architecture and Addressing modes, Memory Segmentation, Instruction Set, Assembly Language Processing, Programming with DOS and BIOS function calls, minimum and maximum mode configuration, Programmable Peripheral Interface (8255), Programmable Timer Controller (8254), Memory Interface to 8086.

Module:3	Microcontroller Architecture: Intel 8051	7 hours

Microcontroller 8051 - Organization and Architecture, RAM-ROM Organization, Machine Cycle, Instruction set: Addressing modes, Data Processing - Stack, Arithmetic, Logical; Branching – Unconditional and Conditional, Assembly programming.

Module:4	Microcontroller 8051 Peripherals	5 hours

I/O Ports, Timers-Counters, Serial Communication and Interrupts.

Module:5	I/O interfacing with Microcontroller 8051	7 hours

LCD, LED, Keypad, Analog-to-Digital Convertors, Digital-to-Analog Convertors, Sensor with Signal Conditioning Interface.

Module:6	ARM Processor Architecture	5 hours

ARM Design Philosophy; Overview of ARM architecture; States [ARM, Thumb, Jazelle]; Registers, Modes; Conditional Execution; Pipelining; Vector Tables; Exception handling.

Module:7	ARM Instruction Set	8 hours

ARM Instruction- data processing instructions, branch instructions, load store instructions, SWI Instruction, Loading instructions, conditional Execution, Assembly Programming.

Module:8	Contemporary issues	2 hours

	Total Lecture hours:	**45 hours**
Text Book(s)		
1.	A.K. Ray, K.M. Bhurchandi, Advanced Microprocessor and Peripherals, 2012, 2nd Edition, Tata McGraw-Hill, India.	
2.	Mohammad Ali Mazidi, Janice G. Mazidi, Rolin D. McKinlay, The 8051 Microcontroller and Embedded Systems, 2014, 2nd Edition, Pearson, India.	
Reference Books		
1.	Muhammad Ali Mazidi, ARM Assembly Language Programming & Architecture: 1, 2016, 2nd Edition, Microdigitaled.com	
2.	A. Nagoor Kani, 8086 Microprocessors and its Applications, 2017, Second Edition, Tata McGraw-Hill Education Pvt. Ltd., New Delhi, India.	
3.	Joseph Yiu, The Definitive Guide to ARM® Cortex®-M0 and Cortex-M0+ Processors, 2015, 2nd Edition, Elsevier Science & Technology, UK	
Mode of Evaluation: Continuous Assessment Test, Digital Assignment, Quiz and Final Assessment Test		
Recommended by Board of Studies	14-05-2022	
Approved by Academic Council	No. 66	Date 16-06-2022

Course Code	Course Title	L	T	P	C
BECE204P	Microprocessors and Microcontrollers Lab	0	0	2	1
Pre-requisite	BECE102L	Syllabus version			
		1.0			

Course Objectives

1. To familiarize the students with assembly language programming using microprocessor and microcontroller.
2. To familiarize the students with Embedded C language programming using microcontroller.
3. To interface peripherals and I/O devices with the microcontroller and microprocessor.

Course Outcome

Student will be able to

1. Showcase the skill, knowledge and ability of programming microcontroller and microprocessor using its instruction set.
2. Expertise with microcontroller and interfaces including general purpose input/ output, timers, serial communication, LCD, keypad and ADC.

Indicative Experiments [Experiments using 8086/8051/ARM]

1	Assembly language programming of Arithmetic/logical operations.	**6 hours**
2	Assembly language programming of memory operations.	**4 hours**
3	Assembly language programming/ Embedded C programming for interfacing the peripherals: General purpose input/ output, timers, serial communication, LCD, keypad and ADC.	**10 hours**
4	Hardware implementation of peripheral interfacing: General purpose input/ output, timers, serial communication, LCD, keypad and ADC.	**10 hours**
	Total Laboratory Hours	**30 hours**

Mode of Assessment: Continuous Assessment and Final Assessment Test			
Recommended by Board of Studies	14-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

BMAT205L	Discrete Mathematics and Graph Theory	L	T	P	C
		3	1	0	4
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives:

1. To address the challenges of the relevance of lattice theoryand algebraic structures to computer science and engineering problems.
2. To use Counting techniques, in particular recurrence relations to computer science problems.
3. To understand the concepts of graph theory and related algorithm concepts.

Course Outcomes:

At the end of this course, students are expected to

1. Learn proof techniques and concepts of inference theory
2. Use algebraic structures in applications
3. Counting techniques in engineering problems.
4. Use lattice and Boolean algebra properties in Digital circuits.
5. Solve Science and Engineering problems using Graph theory.

Module:1	**Mathematical Logic**	**7 hours**

Statements and Notation-Connectives–Tautologies-Equivalence - Implications–Normal forms - The Theory of Inference for the Statement Calculus - Predicate Calculus - Inference Theory of the Predicate Calculus

Module:2	**Algebraic Structures**	**6 hours**

Semigroups and Monoids - Groups – Subgroups – Lagrange's Theorem Homomorphism – Properties-Group Codes.

Module:3	**Counting Techniques**	**6 hours**

Basics of counting - Pigeonhole principle - Permutations and combinations - Inclusion-exclusion principle - Recurrence relations - Solving recurrence relations - Generating functions-Solution to recurrence relations.

Module:4	**Lattices and Boolean algebra**	**6 hours**

Partially Ordered Relations -Lattices as Posets – Hasse Digram – Properties of Lattices – Boolean algebra-Properties of Boolean Algebra-Boolean functions.

Module:5	**Fundamentals of Graphs**	**6hours**

Basic Concepts of Graph Theory – Planar and Complete graph - Matrix representation of Graphs – Graph Isomorphism – Connectivity–Cut sets-Euler and Hamilton Paths–Shortest Path algorithms

Module:6	**Trees, Fundamental circuits, Cut sets**	**6 hours**

Trees – properties of trees – distance and centres in tree – Spanning trees – Spanning tree algorithms- Tree traversals- Fundamental circuits and cut-sets

Module:7	**Graph colouring, covering, Partitioning**	**6 hours**

Bipartite graphs - Chromatic number – Chromatic partitioning – Chromatic polynomial - matching – Covering– Four Colour problem.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**
	Total Tutorial hours:	**15 hours**

Text Books:

1. Discrete Mathematical Structures with Applications to Computer Science, J .P. Trembley and R. Manohar, Tata McGraw Hill-35th reprint, 2017.
2. Graph theory with application to Engineering and Computer Science, NarasingDeo,

Prentice Hall India 2016.			
Reference Books:			
1. Discrete Mathematics and its applications, Kenneth H. Rosen, 8th Edition, Tata McGraw Hill, 2019. 2. Discrete Mathematical Structures, Kolman, R.C.Busby and S.C.Ross, 6th Edition, PHI, 2018. 3. Discrete Mathematics, Richard Johnsonbaugh, 8th Edition, Prentice Hall, 2017. 4. Discrete Mathematics, S. Lipschutz and M. Lipson, McGraw Hill Education (India) 2017. 5. Elements of Discrete Mathematics–A Computer Oriented Approach, C.L.Liu, Tata McGraw Hill, Special Indian Edition, 2017. 6.Introduction to Graph Theory, D. B. West, 3rd Edition, Prentice-Hall, Englewood Cliffs, NJ, 2015.			
Mode of Evaluation: CAT, Quizzes, Digital Assignments, FAT			
Recommended by Board of Studies	15.02.2022		
Approved by Academic Council	No. 65	Date	17-03-2022

Discipline Core

BCSE202L	Data Structures and Algorithms	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart basic concepts of data structures and algorithms.
2. To differentiate linear, non-linear data structures and their operations.
3. To comprehend the necessity of time complexity in algorithms.

Course Outcomes

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

1. Understand the fundamental analysis and time complexity for a given problem.
2. Articulate linear, non-linear data structures and legal operations permitted on them.
3. Identify and apply suitable algorithms for searching and sorting.
4. Discover various tree and graph traversals.
5. Explicate hashing, heaps and AVL trees and realize their applications.

Module:1	**Algorithm Analysis**	**8 hours**

Importance of algorithms and data structures - Fundamentals of algorithm analysis: Space and time complexity of an algorithm, Types of asymptotic notations and orders of growth - Algorithm efficiency – best case, worst case, average case - Analysis of non-recursive and recursive algorithms - Asymptotic analysis for recurrence relation: Iteration Method, Substitution Method, Master Method and Recursive Tree Method.

Module:2	**Linear Data Structures**	**7 hours**

Arrays: 1D and 2D array- Stack - Applications of stack: Expression Evaluation, Conversion of Infix to postfix and prefix expression, Tower of Hanoi – Queue - Types of Queue: Circular Queue, Double Ended Queue (deQueue) - Applications – List: Singly linked lists, Doubly linked lists, Circular linked lists- Applications: Polynomial Manipulation.

Module:3	**Searching and Sorting**	**7 hours**

Searching: Linear Search and binary search – Applications.
Sorting: Insertion sort, Selection sort, Bubble sort, Counting sort, Quick sort, Merge sort - Analysis of sorting algorithms.

Module:4	**Trees**	**6 hours**

Introduction - Binary Tree: Definition and Properties - Tree Traversals- Expression Trees:- Binary Search Trees - Operations in BST: insertion, deletion, finding min and max, finding the k^{th} minimum element.

Module:5	**Graphs**	**6 hours**

Terminology – Representation of Graph – Graph Traversal: Breadth First Search (BFS), Depth First Search (DFS) - Minimum Spanning Tree: Prim's, Kruskal's - Single Source Shortest Path: Dijkstra's Algorithm.

Module:6	**Hashing**	**4 hours**

Hash functions - Separate chaining - Open hashing: Linear probing, Quadratic probing, Double hashing - Closed hashing - Random probing – Rehashing - Extendible hashing.

Module:7	**Heaps and AVL Trees**	**5 hours**

Heaps - Heap sort- Applications -Priority Queue using Heaps. AVL trees: Terminology, basic operations (rotation, insertion and deletion).

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. Mark A. Weiss, Data Structures & Algorithm Analysis in C++, 4^{th} Edition, 2013, Pearson Education.

Reference Books			
1.	Alfred V. Aho, Jeffrey D. Ullman and John E. Hopcroft, Data Structures and Algorithms, 1983, Pearson Education.		
2.	Horowitz, Sahni and S. Anderson-Freed, Fundamentals of Data Structures in C, 2008, 2nd Edition, Universities Press.		
3.	Thomas H. Cormen, C.E. Leiserson, R L. Rivest and C. Stein, Introduction to Algorithms, 2009, 3rd Edition, MIT Press.		
Mode of Evaluation: CAT, Assignment, Quiz and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE202P	Data Structures and Algorithms Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart basic concepts of data structures and algorithms.
2. To differentiate linear, non-linear data structures and their operations.
3. To comprehend the necessity of time complexity in algorithms.

Course Outcomes

On completion of this course, students should be able to:
1. Apply appropriate data structures to find solutions to practical problems.
2. Identify suitable algorithms for solving the given problems.

Indicative Experiments

1.	Implementation of stack data structure and its applications
2.	Implementation of queue data structure and its applications
3.	Implementation linked list and its application
4.	Implementation of searching algorithms
5.	Implementation of sorting algorithms
6.	Binary Tree Traversal implementation
7.	Binary Search Tree implementation
8.	Graph Traversal – Depth First Search and Breadth First Search algorithm
9.	Minimum Spanning Tree – Prim's and Kruskal's algorithm
10.	Single Source Shortest Path Algorithm - Dijkstra's algorithm
	Total Laboratory Hours 30 hours

Text Book

1.	Mark A. Weiss, Data Structures & Algorithm Analysis in C++, 2013, 4th Edition, Pearson.

Reference Books

1.	Alfred V. Aho, Jeffrey D. Ullman and John E. Hopcroft, Data Structures and Algorithms, 1983, Pearson Education.
2.	Horowitz, Sahni and S. Anderson-Freed, Fundamentals of Data Structures in C, 2008, 2nd Edition, Universities Press.
3.	Thomas H. Cormen, C.E. Leiserson, R L. Rivest and C. Stein, Introduction to Algorithms, 2009, 3rd Edition, MIT Press.

Mode of assessment: Continuous assessments and FAT.

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE204L	Design and Analysis of Algorithms	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To provide mathematical foundations for analyzing the complexity of the algorithms
2. To impart the knowledge on various design strategies that can help in solving the real world problems effectively
3. To synthesize efficient algorithms in various engineering design situations

Course Outcomes

On completion of this course, student should be able to:

1. Apply the mathematical tools to analyze and derive the running time of the algorithms
2. Demonstrate the major algorithm design paradigms.
3. Explain major graph algorithms, string matching and geometric algorithms along with their analysis.
4. Articulating Randomized Algorithms.
5. Explain the hardness of real-world problems with respect to algorithmic efficiency and learning to cope with it.

Module:1	**Design Paradigms: Greedy, Divide and Conquer Techniques**	**6 hours**

Overview and Importance of Algorithms - Stages of algorithm development: Describing the problem, Identifying a suitable technique, Design of an algorithm, Derive Time Complexity, Proof of Correctness of the algorithm, Illustration of Design Stages - Greedy techniques: Fractional Knapsack Problem, and Huffman coding - Divide and Conquer: Maximum Subarray, Karatsuba faster integer multiplication algorithm.

Module:2	**Design Paradigms: Dynamic Programming, Backtracking and Branch & Bound Techniques**	**10 hours**

Dynamic programming: Assembly Line Scheduling, Matrix Chain Multiplication, Longest Common Subsequence, 0-1 Knapsack, TSP- Backtracking: N-Queens problem, Subset Sum, Graph Coloring- Branch & Bound: LIFO-BB and FIFO BB methods: Job Selection problem, 0-1 Knapsack Problem

Module:3	**String Matching Algorithms**	**5 hours**

Naïve String-matching Algorithms, KMP algorithm, Rabin-Karp Algorithm, Suffix Trees.

Module:4	**Graph Algorithms**	**6 hours**

All pair shortest path: Bellman Ford Algorithm, Floyd-Warshall Algorithm - Network Flows: Flow Networks, Maximum Flows: Ford-Fulkerson, Edmond-Karp, Push Re-label Algorithm – Application of Max Flow to maximum matching problem

Module:5	**Geometric Algorithms**	**4 hours**

Line Segments: Properties, Intersection, sweeping lines - Convex Hull finding algorithms: Graham's Scan, Jarvis' March Algorithm.

Module:6	**Randomized algorithms**	**5 hours**

Randomized quick sort - The hiring problem - Finding the global Minimum Cut.

Module:7	**Classes of Complexity and Approximation Algorithms**	**7 hours**

The Class P - The Class NP - Reducibility and NP-completeness – SAT (Problem Definition and statement), 3SAT, Independent Set, Clique, Approximation Algorithm – Vertex Cover, Set Cover and Travelling salesman

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009.

Reference Books				
1.	Jon Kleinberg and ÉvaTardos, Algorithm Design, Pearson Education, 1st Edition, 2014.			
2.	Rajeev Motwani, Prabhakar Raghavan; Randomized Algorithms, Cambridge University Press, 1995 (Online Print – 2013)			
3.	Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin, Network Flows: Theory, Algorithms, and Applications, 1st Edition, Pearson Education, 2014.			
Mode of Evaluation: CAT, Written assignments, Quiz, FAT.				
Recommended by Board of Studies	04-03-2022			
Approved by Academic Council	No. 65	Date	17-03-2022	

BCSE204P	Design and Analysis of Algorithms Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**Nil**	**Syllabus version**			
		1.0			

Course Objectives

1. To provide mathematical foundations for analyzing the complexity of the algorithms
2. To impart the knowledge on various design strategies that can help in solving the real world problems effectively
3. Synthesize efficient algorithms in various engineering design situations

Course Outcome

On completion of this course, student should be able to:
1. Demonstrate the major algorithm design paradigms.
2. Explain major graph algorithms, string matching and geometric algorithms along with their analysis.

Indicative Experiments

1.	Greedy Strategy : Activity Selection & Huffman coding
2.	Dynamic Programming : ALS, Matrix Chain Multiplication , Longest Common Subsequence, 0-1 Knapsack
3.	Divide and Conquer : Maximum Subarray and Karatsuba faster integer multiplication algorithm
4.	Backtracking: N-queens
5.	Branch and Bound: Job selection
6	String matching algorithms : Naïve, KMP and Rabin Karp,suffix trees
7	MST and all pair shortest path algorithms
8	Network Flows : Ford –Fulkerson and Edmond - Karp
9	Intersection of line segments &Finding Convexhull, Finding closest pair of points
10	Polynomial time algorithm for verification of NPC problems
11	Approximation and Randomized algorithms
	Total Laboratory Hours 30 Hours

Text Book

1.	Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009.

Reference Books

1.	Jon Kleinberg and ÉvaTardos, Algorithm Design, Pearson Education, 1st Edition, 2014.
2.	Rajeev Motwani, Prabhakar Raghavan; Randomized Algorithms, Cambridge University Press, 1995 (Online Print – 2013)
3.	Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin, Network Flows: Theory, Algorithms, and Applications, 1st Edition, Pearson Education, 2014.

Mode of assessment: Continuous assessments, FAT.

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE205L	Computer Architecture and Organization	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus Version**			
		1.0			

Course Objectives

1. To acquaint students with the basic concepts of fundamental component, architecture, register organization and performance metrics of a computer and to impart the knowledge of data representation in binary and to understand the implementation of arithmetic algorithms in a typical computer.
2. To teach students how to describe machine capabilities and design an effective data path design for instruction execution. To introduce students to syntax and semantics of machine level programming.
3. To make students understand the importance of memory systems, IO interfacing techniques and external storage and their performance metrics for a typical computer. And explore various alternate techniques for improving the performance of a processor.

Course Outcomes

On completion of this course, student should be able to:

1. Differentiate Von Neumann, Harvard, and CISC and RISC architectures. Analyze the performance of machine with different capabilities. Recognize different instruction formats and addressing modes. Validate efficient algorithm for fixed point and floating point arithmetic operations.
2. Explain the importance of hierarchical memory organization. Able to construct larger memories. Analyze and suggest efficient cache mapping technique and replacement algorithms for given design requirements. Demonstrate hamming code for error detection and correction.
3. Understand the need for an interface. Compare and contrast memory mapping and IO mapping techniques. Describe and Differentiate different modes of data transfer. Appraise the synchronous and asynchronous bus for performance and arbitration.
4. Assess the performance of IO and external storage systems. Classify parallel machine models. Analyze the pipeline hazards and solutions.

Module:1	**Introduction To Computer Architecture and Organization**	**5 Hours**

Overview of Organization and Architecture –Functional components of a computer: Registers and register files - Interconnection of components - Overview of IAS computer function - Organization of the von Neumann machine - Harvard architecture - CISC & RISC Architectures.

Module:2	**Data Representation and Computer Arithmetic**	**5 Hours**

Algorithms for fixed point arithmetic operations: Multiplication (Booths, Modified Booths), Division (restoring and non-restoring) - Algorithms for floating point arithmetic operations - Representation of nonnumeric data (character codes).

Module:3	**Instruction Sets and Control Unit**	**9 Hours**

Computer Instructions: Instruction sets, Instruction Set Architecture, Instruction formats, Instruction set categories - Addressing modes - Phases of instruction cycle – ALU - Data-path and control unit: Hardwired control unit and Micro programmed control unit - Performance metrics: Execution time calculation, MIPS, MFLOPS.

Module:4	**Memory System Organization and Architecture**	**7 Hours**

Memory systems hierarchy: Characteristics, Byte Storage methods, Conceptual view of memory cell - Design of scalable memory using RAM's- ROM's chips - Construction of larger size memories - Memory Interleaving - Memory interface address map- Cache memory: principles, Cache memory management techniques, Types of caches, caches misses, Mean

memory access time evaluation of cache.		
Module:5	**Interfacing and Communication**	**5 Hours**
I/O fundamentals: handshaking, buffering, I/O Modules - I/O techniques: Programmed I/O, Interrupt-driven I/O, Direct Memory Access, Direct Cache Access - Interrupt structures: Vectored and Prioritized-interrupt overhead - Buses: Synchronous and asynchronous - Arbitration.		
Module:6	**Subsystems**	**5 Hours**
External storage systems: Solid state drivers - Organization and Structure of disk drives: Electronic- magnetic and optical technologies - Reliability of memory systems - Error detecting and error correcting systems - RAID Levels - I/O Performance		
Module:7	**High Performance Processors**	**7 Hours**
Classification of models - Flynn's taxonomy of parallel machine models (SISD, SIMD, MISD, MIMD) - Pipelining: Two stages, Multi stage pipelining, Basic performance issues in pipelining, Hazards, Methods to prevent and resolve hazards and their drawbacks - Approaches to deal branches - Superscalar architecture: Limitations of scalar pipelines, superscalar versus super pipeline architecture, superscalar techniques, performance evaluation of superscalar architecture - performance evaluation of parallel processors: Amdahl's law, speed-up and efficiency.		
Module:8	**Contemporary Issues**	**2 Hours**
	Total Lecture Hours	**45 Hours**

Text Book(s)	
1	David A. Patterson and John L. Hennessy, Computer Organization and Design -The Hardware / Software Interface 6th Edition, Morgan Kaufmann, 2020
Reference Book(s)	
1	Computer Architecture and Organization-Designing for Performance, William Stallings, Tenth edition, Pearson Education series, 2016
2	Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization, Mc Graw Hill, Fifth edition, Reprint 2011.
Mode of Evaluation: CAT, Written Assignments, Quiz and FAT.	

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE301L	Software Engineering	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the essential Software Engineering concepts.
2. To impart concepts and skills for performing analysis, design ,develop, test and evolve efficient software systems of various disciplines and applications
3. To make familiar about engineering practices, standards and metrics for developing software components and products.

Course Outcomes

On completion of this course, student should be able to:

1. Apply and assess the principles of various process models for the software development.
2. Demonstrate various software project management activities that include planning , Estimations, Risk assessment and Configuration Management
3. Perform Requirements modelling and apply appropriate design and testing heuristics to produce quality software systems.
4. Demonstrate the complete Software life cycle activities from requirements analysis to maintenance using the modern tools and techniques.
5. Escalate the use of various standards and metrics in evaluating the process and product.

Module:1	**Overview Of Software Engineering**	**6 hours**

Nature of Software, Software Engineering, Software process, project, product, Process Models

Classical Evolutionary models, Introduction to Agility - Agile Process-Extreme programming - XP Process – Principles of Agile Software Development framework - Overview of System Engineering

Module:2	**Introduction To Software Project Management**	**6 hours**

Planning, Scope, Work break-down structure, Milestones, Deliverables, Cost and Estimates - (Human Resources, Time-scale, Costs), Risk Management, RMMM Plan, CASE TOOLS, Agile Project Management, Managing team dynamics and communication, Metrics and Measurement

Module:3	**Modelling Requirements**	**8 hours**

Software requirements and its types, Requirements Engineering process, Requirement Elicitation, System Modeling – Requirements Specification and Requirement Validation, Requirements Elicitation techniques, Requirements management in Agile.

Module:4	**Software Design**	**8 hours**

Design concepts and principles - Abstraction - Refinement - Modularity Cohesion coupling, Architectural design, Detailed Design Transaction Transformation, Refactoring of designs, Object oriented Design User-Interface Design

Module:5	**Validation And Verification**	**7 hours**

Strategic Approach to Software Testing, Testing Fundamentals Test Plan, Test Design, Test Execution, Reviews, Inspection and Auditing – Regression Testing – Mutation Testing - Object oriented testing - Testing Web based System - Mobile App testing – Mobile test Automation and tools – DevOps Testing – Cloud and Big Data Testing

Module:6	**Software Evolution**	**4 hours**

Software Maintenance, Types of Maintenance, - Software Configuration Management – Overview – SCM Tools. Re-Engineering, Reverse Engineering, Software Reuse		
Module:7	**Quality Assurance**	**4 hours**
Product and Process Metrics, Quality Standards Models ISO, TQM, Six-Sigma, Process improvement Models: CMM & CMMI. Quality Control and Quality Assurance - Quality Management - Quality Factors - Methods of Quality Management		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Ian Somerville, Software Engineering, 10th Edition, Addison-Wesley, 2015
Reference Books	
1.	Roger S. Pressman and Bruce R. Maxim, Software Engineering: A Practitioner's Approach, 10th edition, McGraw Hill Education, 2019
2.	William E. Lewis , Software Testing and Continuous Quality Improvement, Third Edition, Auerbach Publications, 2017

Mode of Evaluation: CAT, Written assignment, Quiz, FAT.			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE301P	Software Engineering Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the essential Software Engineering concepts.
2. To impart concepts and skills for performing analysis, design ,develop, test and evolve efficient software systems of various disciplines and applications
3. To make familiar about engineering practices, standards and metrics for developing software components and products.

Course Outcome

On completion of this course, student should be able to:

1. Demonstrate the complete Software life cycle activities from requirements analysis to maintenance using the modern tools and techniques.

Indicative Experiments	
1.	Analysis and Identification of the suitable process models
2.	Work Break-down Structure (Process Based, Product Based, Geographic Based and Role Based) and Estimations
3.	Requirement modelling using Entity Relationship Diagram(Structural Modeling)
4.	Requirement modelling using Context flow diagram, DFD (Functional Modeling)
5.	Requirement modelling using State Transition Diagram (Behavioral Modeling)
6.	OO design – Use case Model, Class Model
7.	OO design – Interaction Models
8.	OO design – Package, Component and deployment models
9.	Design and demonstration of test cases. Functional Testing and Non- Functional Testing (using any open source tools)
10.	Story Boarding and User Interface design Modelling
	Total Laboratory Hours: 30 hours

Text Book(s)

1. Ian Somerville, Software Engineering, 10th Edition, Addison-Wesley, 2015

Reference Books

1. Roger S. Pressman and Bruce R. Maxim, Software Engineering: A Practitioner's Approach, 10th edition, McGraw Hill Education, 2019
2. William E. Lewis, Software Testing and Continuous Quality Improvement, Third Edition, Auerbach Publications, 2017

Mode of assessment: Continuous assessments, FAT.

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE302L	Database Systems	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the concepts of File system and structure of the database; Designing an Entity-Relationship model for a real-life application and Mapping a database schema from the ER model.
2. To differentiate various normal forms, evaluate relational schemas for design qualities and optimize a query.
3. To impart the working methodologies of transaction management, understand concurrency control, recovery, indexing, access methods and fundamental view on unstructured data and its management.

Course Outcomes

On completion of this course, student should be able to:

1. Comprehend the role of database management system in an organization and design the structure and operation of the relational data model.
2. Develop a database project depending on the business requirements, considering various design issues.
3. List the concepts of indexing and accessing methods.
4. Explain the concept of a database transaction processing and comprehend the concept of database facilities including concurrency control, backup and recovery.
5. Review the fundamental view on unstructured data and describe other emerging database technologies.

Module:1	**Database Systems Concepts and Architecture**	**4 hours**

Need for database systems – Characteristics of Database Approach – Advantages of using DBMS approach - Actors on the Database Management Scene: Database Administrator - Classification of database management systems - Data Models - Schemas and Instances - Three-Schema Architecture - The Database System Environment - Centralized and Client/Server Architectures for DBMSs – Overall Architecture of Database Management Systems

Module:2	**Relational Model and E-R Modeling**	**6 hours**

Relational Model: Candidate Keys, Primary Keys, Foreign Keys - Integrity Constraints - Handling of Nulls - Entity Relationship Model: Types of Attributes, Relationships, Structural Constraints, Relational model Constraints – Mapping ER model to a relational schema – Extended ER Model - Generalization – Specialization – Aggregations.

Module:3	**Relational Database Design**	**6 hours**

Database Design – Schema Refinement - Guidelines for Relational Schema - Functional dependencies - Axioms on Functional Dependencies- Normalization: First, Second and Third Normal Forms - Boyce Codd Normal Form, Multi-valued dependency and Fourth Normal form - Join dependency and Fifth Normal form

Module:4	**Physical Database Design and Query Processing**	**8 hours**

File Organization - Indexing: Single level indexing, multi-level indexing, dynamic multilevel Indexing - B+ Tree Indexing – Hashing Techniques: Static and Dynamic Hashing – Relational Algebra - Translating SQL Queries into Relational Algebra - Query Processing – Query Optimization: Algebraic Query Optimization, Heuristic query optimization Rules, Join Query Optimization using Indexing and Hashing - Tuple Relational Calculus.

Module:5	**Transaction Processing and Recovery**	**8 hours**

Introduction to Transaction Processing – Transaction concepts: ACID Properties of Transactions, Transaction States - Serial and Serializable Schedules - Schedules based on recoverability – Schedules based on Serializability - Conflict Serializabilty - Recovery Concepts: Log Based Recovery Protocols, Recovery based on deferred update, Recovery techniques based on immediate update – Shadow Paging Algorithm		
Module:6	**Concurrency Control In Transaction Processing**	**8 hours**
Concurrent Transactions – Lost Update Problem - Concurrency Control Techniques: Time Stamp Based Protocols, Thomas Write Rule, Lock Based Protocols, Lock Compatibility Matrix, - Two-Phase Locking Protocol - Lock Conversions - Graph Based Protocols for Concurrency Control - Tree Protocol for Concurrency Control – Deadlocks Based on Locks in Transactions – Deadlock Handling Techniques – Transaction Deadlock Detection Techniques – Transaction Deadlock Prevention Techniques – Multi-Granularity Locking for avoiding Transaction Deadlocks		
Module:7	**NOSQL Database Management**	**3 hours**
Introduction, Need of NoSQL, CAP Theorem, different NoSQL data bases: Key-value data stores, Columnar families, Document databases, Graph databases		
Module:8	**Contemporary Issues**	**2 Hours**
	Total Lecture hours:	**45 hours**

Text Book	
1.	R. Elmasri & S. B. Navathe, Fundamentals of Database Systems, Addison Wesley, 7th Edition, 2016
Reference Books	
1.	A. Silberschatz, H. F. Korth & S. Sudarshan, Database System Concepts, McGraw Hill, 7th Edition 2019.
2.	Raghu Ramakrishnan, Database Management Systems, Mcgraw-Hill, 4th Edition, 2018
3.	C.J.Date, A.Kannan, S.Swamynathan," An Introduction to Database Systems", Pearson, Eighth Edition, 2006.
4.	Gerardus Blokdyk, NoSQL Databases A Complete Guide, 5STARCooks, 2021
Mode of Evaluation: CAT, Written assignments, Quiz and FAT.	

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE302P	Database Systems Lab	L	T	P	C
		0	0	2	1
Pre-requisite		**Syllabus version**			
		1.0			

Course Objectives

1. Basic ability to understand the concepts of File system and structure of the database; Designing an Entity-Relationship model for a real-life application and Mapping a database schema from the ER model.
2. Differentiate various normal forms, evaluate relational schemas for design qualities and optimize a query.
3. Explain the working methodologies of transaction management and give a solution during a transaction failure. Understand the basic concepts on concurrency control, recovery, indexing, access methods and fundamental view on unstructured data and its management.

Course Outcome

On completion of this course, student should be able to:

1. Design the structure and operation of the relational data model.
2. Examine the data requirements of the real world and design a database management system.

Indicative Experiments	
1.	Data Definition and Data Manipulation Language
2.	Constraints
3.	Single row functions
4.	Operators and group functions
5.	Sub query, views and joins
6.	High Level Language Extensions - Procedures, Functions, Cursors and Triggers
	Total Laboratory Hours 30 hours

Text Book

1. R. Elmasri & S. B. Navathe, Fundamentals of Database Systems, Addison Wesley, 7th Edition, 2016

Reference Books

1. A. Silberschatz, H. F. Korth & S. Sudarshan, Database System Concepts, McGraw Hill, 7th Edition 2019.
2. Raghu Ramakrishnan, Database Management Systems, Mcgraw-Hill, 4th Edition, 2018
3. C.J.Date, A.Kannan, S.Swamynathan," An Introduction to Database Systems", Pearson, Eighth Edition, 2006.
4. Gerardus Blokdyk, NoSQL Databases A Complete Guide, 5STARCooks, 2021

Mode of assessment: Continuous assessments, FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE303L	Operating Systems	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the operating system concepts, designs and provide skills required to implement the services.
2. To describe the trade-offs between conflicting objectives in large scale system design.
3. To develop the knowledge for application of the various design issues and services.

Course Outcomes

On completion of this course, student should be able to:

1. Interpret the evolution of OS functionality, structures, layers and apply various types of system calls of various process states.
2. Design scheduling algorithms to compute and compare various scheduling criteria.
3. Apply and analyze communication between inter process and synchronization techniques.
4. Implement page replacement algorithms, memory management problems and segmentation.
5. Differentiate the file systems for applying different allocation, access technique, representing virtualization and providing protection and security to OS.

Module:1	**Introduction**	**3 hours**

Introduction to OS: Functionality of OS - OS design issues - Structuring methods (monolithic, layered, modular, micro-kernel models) - Abstractions, processes, resources - Influence of security, networking, and multimedia.

Module:2	**OS Principles**	**4 hours**

System calls, System/Application Call Interface – Protection: User/Kernel modes - Interrupts -Processes - Structures (Process Control Block, Ready List etc.), Process creation, management in Unix – Threads: User level, kernel level threads and thread models.

Module:3	**Scheduling**	**9 hours**

Processes Scheduling - CPU Scheduling: Pre-emptive, non-pre-emptive - Multiprocessor scheduling – Deadlocks - Resource allocation and management - Deadlock handling mechanisms: prevention, avoidance, detection, recovery.

Module:4	**Concurrency**	**8 hours**

Inter-process communication, Synchronization - Implementing synchronization primitives (Peterson's solution, Bakery algorithm, synchronization hardware) - Semaphores – Classical synchronization problems, Monitors: Solution to Dining Philosophers problem – IPC in Unix, Multiprocessors and Locking - Scalable Locks - Lock-free coordination.

Module:5	**Memory Management**	**7 hours**

Main memory management, Memory allocation strategies, Virtual memory: Hardware support for virtual memory (caching, TLB) – Paging - Segmentation - Demand Paging - Page Faults - Page Replacement -Thrashing - Working Set.

Module:6	**Virtualization and File System Management**	**6 hours**

Virtual Machines - Virtualization (Hardware/Software, Server, Service, Network - Hypervisors - Container virtualization - Cost of virtualization - File system interface (access methods, directory structures) - File system implementation (directory implementation, file allocation methods) - File system recovery - Journaling - Soft updates - Log-structured file system - Distributed file system.

Module:7	**Storage Management, Protection and Security**	**6 hours**

Disk structure and attachment – Disk scheduling algorithms (seek time, rotational latency based)- System threats and security – Policy vs mechanism - Access vs authentication -

System protection: Access matrix – Capability based systems - OS: performance, scaling, future directions in mobile OS.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**
Text Book		
1.	Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", 2018, 10th Edition, Wiley, United States.	
Reference Books		
1.	Andrew S. Tanenbaum, "Modern Operating Systems", 2016, 4th Edition, Pearson, United Kingdom.	
2.	William Stallings, "Operating Systems: Internals and Design Principles", 2018, 9th Edition, Pearson, United Kingdom.	
Mode of Evaluation: CAT, Written Assignment, Quiz, FAT		
Recommended by Board of Studies	04-03-2022	
Approved by Academic Council	No. 65	Date 17-03-2022

BCSE303P	Operating Systems Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**Nil**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the operating system concepts, designs and provide skills required to implement the services.
2. To describe the trade-offs between conflicting objectives in large scale system design.
3. To develop the knowledge for application of the various design issues and services.

Course Outcome

On completion of this course, student should be able to:

1. Interpret the evolution of OS functionality, structures, layers and apply various types of system calls of various process states.
2. Design scheduling algorithms to compute and compare various scheduling criteria.
3. Apply and analyze communication between inter process and synchronization techniques.
4. Implement page replacement algorithms, memory management problems and segmentation.
   Differentiate the file systems for applying different allocation, access technique, representing virtualization and providing protection and security to OS.

Indicative Experiments

1.	Study of Basic Linux Commands
2.	Implement your own bootloader program that helps a computer to boot an OS.
3.	Shell Programming (I/O, Decision making, Looping, Multi-level branching)
4.	Creating child process using fork () system call, Orphan and Zombie process creation
5.	Simulation of CPU scheduling algorithms (FCFS, SJF, Priority and Round Robin)
6.	Implement process synchronization using semaphores / monitors.
7.	Simulation of Banker s algorithm to check whether the given system is in safe state or not. Also check whether addition resource requested can be granted immediately
8.	Parallel Thread management using Pthreads library. Implement a data parallelism using multi-threading
9.	Dynamic memory allocation algorithms - First-fit, Best-fit, Worst-fit algorithms
10.	Page Replacement Algorithms FIFO, LRU and Optimal
11.	Implement a file locking mechanism.
12.	Virtualization Setup: Type-1, Type-2 Hypervisor (Detailed Study Report)
	Total Laboratory Hours 30 hours

Text Book

1. Fox, Richard, “Linux with Operating System Concepts”, 2022, 2nd Edition, Chapman and Hall/CRC, UK.

Reference Books

1. Love, Robert, "Linux System Programming: talking directly to the kernel and C library", 2013, 2nd Edition, O'Reilly Media, Inc, United States.
2. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, “Operating System Concepts”, 2018, 10th Edition, Wiley, United States.

Mode of Assessment: Continuous Assessments, FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE304L	Theory of Computation	L	T	P	C
		3	0	0	3
Pre-requisite	**Nil**	**Syllabus version**			
		1.0			

Course Objectives

1. Types of grammars and models of automata.
2. Limitation of computation: What can be and what cannot be computed.
3. Establishing connections among grammars, automata and formal languages.

Course Outcome

On completion of this course, student should be able to:
1. Compare and analyse different computational models
2. Apply rigorously formal mathematical methods to prove properties of languages, grammars and automata.
3. Identify limitations of some computational models and possible methods of proving them.
4. Represent the abstract concepts mathematically with notations.

Module:1	**Introduction to Languages and Grammars**	**4 hours**

Recall on Proof techniques in Mathematics - Overview of a Computational Models - Languages and Grammars - Alphabets - Strings - Operations on Languages, Overview on Automata

Module:2	**Finite State Automata**	**8 hours**

Finite Automata (FA) - Deterministic Finite Automata (DFA) - Non-deterministic Finite Automata (NFA) - NFA with epsilon transitions – NFA without epsilon transition, conversion of NFA to DFA, Equivalence of NFA and DFA – minimization of DFA

Module:3	**Regular Expressions and Languages**	**7 hours**

Regular Expression - FA and Regular Expressions: FA to regular expression and regular expression to FA - Pattern matching and regular expressions - Regular grammar and FA - Pumping lemma for regular languages - Closure properties of regular languages

Module:4	**Context Free Grammars**	**7 hours**

Context-Free Grammar (CFG) – Derivations - Parse Trees - Ambiguity in CFG - CYK algorithm – Simplification of CFG – Elimination of Useless symbols, Unit productions, Null productions - Normal forms for CFG: CNF and GNF - Pumping Lemma for CFL - Closure Properties of CFL

Module:5	**Pushdown Automata**	**5 hours**

Definition of the Pushdown automata - Languages of a Pushdown automata – Power of Non-Deterministic Pushdown Automata and Deterministic pushdown automata

Module:6	**Turing Machine**	**6 hours**

Turing Machines as acceptor and transducer - Multi head and Multi tape Turing Machines – Universal Turing Machine - The Halting problem - Turing-Church thesis

Module:7	**Recursive and Recursively Enumerable Languages**	**6 hours**

Recursive and Recursively Enumerable Languages, Language that is not Recursively Enumerable (RE) – computable functions – Chomsky Hierarchy – Undecidable problems - Post's Correspondence Problem

Module:8	**Contemporary Issues**	**2 hours**

	Total Lecture hours:	**45 hours**

Text Book

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, "Introduction to Automata Theory, Languages and Computation", Third Edition, Pearson Education, India 2008. ISBN: 978-8131720479

Reference Books

1.	Peter Linz, “An Introduction to Formal Languages and Automata”, Sixth Edition, Jones & Bartlett, 2016. ISBN: 978-9384323219		
2.	K. Krithivasan and R. Rama, “Introduction to Formal Languages, Automata and Computation”, Pearson Education, 2009. ISBN: 978-8131723562		
Mode of Evaluation: CAT, Assignment, Quiz, FAT.			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE305L	Embedded Systems	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To expose students to various challenges and constraints of special purpose computing systems in terms of resources and functional requirements.
2. To introduce students to various components of typical embedded systems viz., sensors and actuators, data converters, UART etc., their interfacing, programming environment for developing any smart systems and various serial communication protocols for optimal components interfacing and communication.
3. To make students understand the importance of program modeling, optimization techniques and debugging tools for product development and explore various solutions for real time scheduling issues in terms of resources and deadline.

Course Outcomes

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

1. Identify the challenges in designing an embedded system using various microcontrollers and interfaces.
2. To summaries the functionality of any special purpose computing system, and to propose smart solutions to engineering challenges at the prototype level.
3. To examine the working principle and interface of typical embedded system components, create programme models, apply various optimization approaches including simulation environment and demonstration using debugging tools.
4. To evaluate the working principle of serial communication protocols and their proper use, as well as to analyze the benefits and drawbacks of real-time scheduling algorithms and to recommend acceptable solutions for specific challenges.

Module:1	**Introduction**	**5 hours**
Overview of Embedded Systems, Design challenges, Embedded processor technology, Hardware Design, Micro-controller architecture -8051, PIC, and ARM.		
Module:2	**I/O Interfacing Techniques**	**8 hours**
Memory interfacing, A/D, D/A, Timers, Watch-dog timer, Counters, Encoder & Decoder, UART, Sensors and actuators interfacing.		
Module:3	**Architecture of Special Purpose Computing System**	**6 hours**
ATM, Handheld devices, Data Compressor, Image Capturing Devices–Architecture and Requirements, Challenges & Constraints of special purpose computing system.		
Module:4	**Programming Tools**	**7 hours**
Evolution of embedded programming tools, Modelling programs, Code optimization, Logic analyzers, Programming environment.		
Module:5	**Real Time Operating System**	**8 hours**
Classification of Real time system, Issues & challenges in RTS, Real time scheduling schemes- EDF-RMS & Hybrid techniques, eCOS, POSIX, Protothreads.		
Module:6	**Embedded Networking Protocols**	**5 hours**
Inter Integrated Circuits (I2C), Controller Area Network, Embedded Ethernet Controller, RS232, Bluetooth, Zigbee, Wifi.		
Module:7	**Applications of Embedded Systems**	**4 hours**
Introduction to embedded system applications using case studies – Role in Agriculture sector, Automotive electronics, Consumer Electronics, Industrial controls, Medical Electronics.		
Module:8	**Contemporary Issues**	**2 hours**

	Total Lecture hours:	**45 hours**
Text Book		
1.	Marilyn Wolf, Computers as Components – Principles of Embedded Computing System Design, Fourth Edition, Morgan Kaufman Publishers, 2016.	
Reference Books		
1.	Embedded Systems Architecture, Programming and Design, by Raj Kamal, McGraw Hill Education, 3e, 2015.	
2.	Embedded System Design A Unified Hardware/Sofware Introduction, by Vahid G Frank and Givargis Tony, John Wiley & Sons, 2009.	
Mode of Evaluation: CAT, written assignment, Quiz, FAT.		
Recommended by Board of Studies	04-03-2022	
Approved by Academic Council	No. 65	Date 17-03-2022

BCSE306L	Artificial Intelligence	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart artificial intelligence principles, techniques and its history.
2. To assess the applicability, strengths, and weaknesses of the basic knowledge representation, problem solving, and learning methods in solving engineering problems
3. To develop intelligent systems by assembling solutions to concrete computational problems

Course Outcomes

On completion of this course, student should be able to:

1. Evaluate Artificial Intelligence (AI) methods and describe their foundations.
2. Apply basic principles of AI in solutions that require problem-solving, inference, perception, knowledge representation and learning.
3. Demonstrate knowledge of reasoning, uncertainty, and knowledge representation for solving real-world problems
4. Analyse and illustrate how search algorithms play a vital role in problem-solving

Module:1	**Introduction**	**6 hours**
Introduction- Evolution of AI, State of Art -Different Types of Artificial Intelligence- Applications of AI-Subfields of AI-Intelligent Agents- Structure of Intelligent Agents- Environments		
Module:2	**Problem Solving based on Searching**	**6 hours**
Introduction to Problem Solving by searching Methods-State Space search, Uninformed Search Methods – Uniform Cost Search, Breadth First Search- Depth First Search-Depth-limited search, Iterative deepening depth-first, Informed Search Methods- Best First Search, A* Search		
Module 3	**Local Search and Adversarial Search**	**5 hours**
Local Search algorithms – Hill-climbing search, Simulated annealing, Genetic Algorithm, Adversarial Search: Game Trees and Minimax Evaluation, Elementary two-players games: tic-tac-toe, Minimax with Alpha-Beta Pruning.		
Module:4	**Logic and Reasoning**	**8 hours**
Introduction to Logic and Reasoning -Propositional Logic-First Order Logic-Inference in First Order Logic- Unification, Forward Chaining, Backward Chaining, Resolution.		
Module:5	**Uncertain Knowledge and Reasoning**	**5 hours**
Quantifying Uncertainty- Bayes Rule -Bayesian Belief Network- Approximate Inference in Bayesian networks		
Module:6	**Planning**	**7 hours**
Classical planning, Planning as State-space search, Forward search, backward search, Planning graphs, Hierarchical Planning, Planning and acting in Nondeterministic domains – Sensor-less Planning, Multiagent planning		
Module:7	**Communicating, Perceiving and Acting**	**6 hours**
Communication-Fundamentals of Language -Probabilistic Language Processing -Information Retrieval- Information Extraction-Perception-Image Formation- Object Recognition.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1.	Russell, S. and Norvig, P. 2015. Artificial Intelligence - A Modern Approach, 3rd Edition, Prentice Hall.

Reference Books			
1.	K. R. Chowdhary, Fundamentals of Artificial Intelligence, Springer, 2020.		
2	Alpaydin, E. 2010. Introduction to Machine Learning. 2nd Edition, MIT Press.		
Mode of Evaluation: CAT, Assignment, Quiz, FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE307L	Compiler Design	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To provide fundamental knowledge of various language translators.
2. To make students familiar with lexical analysis and parsing techniques.
3. To understand the various actions carried out in semantic analysis.
4. To make the students get familiar with how the intermediate code is generated.
5. To understand the principles of code optimization techniques and code generation.
6. To provide foundation for study of high-performance compiler design.

Course Outcomes

1. Apply the skills on devising, selecting, and using tools and techniques towards compiler design
2. Develop language specifications using context free grammars (CFG).
3. Apply the ideas, the techniques, and the knowledge acquired for the purpose of developing software systems.
4. Constructing symbol tables and generating intermediate code.
5. Obtain insights on compiler optimization and code generation.

Module:1	**INTRODUCTION TO COMPILATION AND LEXICAL ANALYSIS**	**7 hours**

Introduction to LLVM - Structure and Phases of a Compiler-Design Issues-Patterns-Lexemes-Tokens-Attributes-Specification of Tokens-Extended Regular Expression- Regular expression to Deterministic Finite Automata (Direct method) - Lex - A Lexical Analyzer Generator.

Module:2	**SYNTAX ANALYSIS**	**8 hours**

Role of Parser- Parse Tree - Elimination of Ambiguity – Top Down Parsing - Recursive Descent Parsing - LL (1) Grammars – Shift Reduce Parsers- Operator Precedence Parsing - LR Parsers, Construction of SLR Parser Tables and Parsing- CLR Parsing- LALR Parsing.

Module:3	**SEMANTICS ANALYSIS**	**5 hours**

Syntax Directed Definition – Evaluation Order - Applications of Syntax Directed Translation - Syntax Directed Translation Schemes - Implementation of L-attributed Syntax Directed Definition.

Module:4	**INTERMEDIATE CODE GENERATION**	**5 hours**

Variants of Syntax trees - Three Address Code- Types – Declarations - Procedures - Assignment Statements - Translation of Expressions - Control Flow - Back Patching- Switch Case Statements.

Module:5	**CODE OPTIMIZATION**	**6 hours**

Loop optimizations- Principal Sources of Optimization -Introduction to Data Flow Analysis - Basic Blocks - Optimization of Basic Blocks - Peephole Optimization- The DAG Representation of Basic Blocks -Loops in Flow Graphs - Machine Independent Optimization- Implementation of a naïve code generator for a virtual Machine- Security checking of virtual machine code.

Module:6	**CODE GENERATION**	**5 hours**

Issues in the design of a code generator- Target Machine- Next-Use Information - Register Allocation and Assignment- Runtime Organization- Activation Records.

Module:7	**PARALLELISM**	**7 hours**

Parallelization- Automatic Parallelization- Optimizations for Cache Locality and Vectorization- Domain Specific Languages-Compilation- Instruction Scheduling and Software Pipelining- Impact of Language Design and Architecture Evolution on Compilers-Static Single Assignment

Module:8	**Contemporary Issues**	**2 hours**

	Total Lecture hours:	**45 hours**
Text Book(s)		
1.	A. V. Aho, Monica S. Lam, Ravi Sethi and Jeffrey D. Ullman, Compilers: Principles, techniques, & tools, 2007, Second Edition, Pearson Education, Boston.	
Reference Books		
1.	Watson, Des. A Practical Approach to Compiler Construction. Germany, Springer International Publishing, 2017.	

Mode of Evaluation: CAT, Quiz, Written assignment and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE307P	Compiler Design Lab	L	T	P	C
		0	0	2	1
Pre-requisite		**Syllabus version**			
		1.0			

Course Objectives

1. To provide fundamental knowledge of various language translators.
2. To make students familiar with phases of compiler.
3. To provide foundation for study of high-performance compiler design.

Course Outcome

1. Apply the skills on devising, selecting and using tools and techniques towards compiler design
2. Develop language specifications using context free grammars (CFG).
3. Apply the ideas, the techniques, and the knowledge acquired for the purpose of developing software systems.
4. Constructing symbol tables and generating intermediate code.
5. Obtain insights on compiler optimization and code generation.

Indicative Experiments	
1.	Implementation of LEXR using LLVM.
2.	Implementation of handwritten parser using LLVM
3.	Generating code with the LLVM backend.
4.	Defining a real programming language.
5.	Write a recursive descent parser for the CFG language and implement it using LLVM.
6.	Write a LR parser for the CFG language and implement it in the using LLVM.
7.	Intro to Flex and Bison Modify the scanner and parser so that terminating a statement with "; b" instead of ";" results in the output being printed in binary.
8.	Using LLVM-style RTTI for the AST and Generating IR from the AST.
9.	Converting types from an AST description to LLVM types.
10.	Emitting assembler text and object code.
	Total Laboratory Hours **30 hours**

Mode of assessment: CAT, FAT

Text Book(s)

1	Learn LLVM 12: A beginner's guide to learning LLVM compiler tools and core libraries with C++

Reference Books

1.	Watson, Des. A Practical Approach to Compiler Construction. Germany, Springer International Publishing, 2017.

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE308L	Computer Networks	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To build an understanding among students about the fundamental concepts of computer networking, protocols, architectures, and applications.
2. To help students to acquire knowledge in design, implement and analyze performance of OSI and TCP-IP based Architectures.
3. To identify the suitable application layer protocols for specific applications and its respective security mechanisms.

Course Outcomes

On completion of this course, student should be able to:

1. Interpret the different building blocks of Communication network and its architecture.
2. Contrast different types of switching networks and analyze the performance of network
3. Identify and analyze error and flow control mechanisms in data link layer.
4. Design sub-netting and analyze the performance of network layer with various routing protocols.
5. Compare various congestion control mechanisms and identify appropriate transport layer protocol for real time applications with appropriate security mechanism.

Module:1	**Networking Principles and Layered Architecture**	**6 hours**
Data Communications and Networking: A Communications Model – Data Communications - Evolution of network, Requirements , Applications, Network Topology (Line configuration, Data Flow), Protocols and Standards, Network Models (OSI, TCP/IP)		
Module:2	**Circuit and Packet Switching**	**7 hours**
Switched Communications Networks – Circuit Switching – Packet Switching – Comparison of Circuit Switching and Packet Switching – Implementing Network Software, Networking Parameters(Transmission Impairment, Data Rate and Performance)		
Module:3	**Data Link Layer**	**8 hours**
Error Detection and Correction – Hamming Code , CRC, Checksum- Flow control mechanism – Sliding Window Protocol - GoBack - N - Selective Repeat - Multiple access Aloha - Slotted Aloha - CSMA, CSMA/CD – IEEE Standards(IEEE802.3 (Ethernet), IEEE802.11(WLAN))- RFID- Bluetooth Standards		
Module:4	**Network Layer**	**8 hours**
IPV4 Address Space – Notations – Classful Addressing – Classless Addressing – Network Address Translation – IPv6 Address Structure – IPv4 and IPv6 header format		
Module:5	**Routing Protocols**	**6 hours**
Routing-Link State and Distance Vector Routing Protocols- Implementation-Performance Analysis- Packet Tracer		
Module:6	**Transport Layer**	**5 hours**
TCP and UDP-Congestion Control-Effects of Congestion-Traffic Management-TCP Congestion Control-Congestion Avoidance Mechanisms-Queuing Mechanisms-QoS Parameters		
Module:7	**Application layer**	**3 hours**
Application layer-Domain Name System-Case Study : FTP-HTTP-SMTP-SNMP		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1.	Behrouz A. Forouzan, Data communication and Networking, 5th Edition, 2017,

	McGraw Hill Education.		
Reference Books			
1.	James F. Kurose and Keith W.Ross, Computer Networking: A Top-Down Approach, 6th Edition, 2017, Pearson Education.		
2.	William Stallings, “Data and Computer Communication”, 10th Edition, 2017, Pearson, United Kingdom.		
Mode of Evaluation: CAT, Written Assignment, Quiz, FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE308P	Computer Networks Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To build an understanding among students about the fundamental concepts of computer networking, protocols, architectures, and applications.
2. To help students to acquire knowledge in design, implement and analyze performance of OSI and TCP-IP based Architectures.
3. To identify the suitable application layer protocols for specific applications and its respective security mechanisms

Course Outcome

On completion of this course, student should be able to:

1. Interpret the different building blocks of Communication network and its architecture.
2. Contrast different types of switching networks and analyze the performance of network
3. Identify and analyze error and flow control mechanisms in data link layer.
4. Design sub-netting and analyze the performance of network layer with various routing protocols.
5. Compare various congestion control mechanisms and identify appropriate transport layer protocol for real time applications with appropriate security mechanism.

Indicative Experiments	
1.	Study of Basic Network Commands, Demo session of all networking hardware and Functionalities
2.	Error detection and correction mechanisms
3.	Flow control mechanisms
4.	IP addressing Classless addressing
5.	Observing Packets across the network and Performance Analysis of Routing protocols
6.	Socket programming(TCP and UDP) - Some challenging experiments can be given on Socket programming
7.	Simulation of unicast routing protocols
8.	Simulation of Transport layer Protocols and analysis of congestion control techniques in network
9.	Develop a DNS client server to resolve the given host name or IP address
	Total Laboratory Hours 30 hours

Text book

1	W.Richard Stevens, Uix Network Programming, 2ndEdition, Pearson Education, 2015.

Mode of assessment: Continuous assessment, FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE309L	Cryptography and Network Security	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To explore the concepts of basic number theory and cryptographic techniques.
2. To impart concept of Hash and Message Authentication, Digital Signatures and authentication protocols.
3. To reveal the basics of transport layer security, Web Security and various types of System Security.

Course Outcomes

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

1. To know the fundamental mathematical concepts related to security.
2. To understand concept of various cryptographic techniques.
3. To apprehend the authentication and integrity process of data for various applications
4. To know fundamentals of Transport layer security, web security, E-Mail Security and IP Security

Module:1	**Fundamentals of Number Theory**	**5 hours**
Finite Fields and Number Theory: Modular arithmetic, Euclidian Algorithm, Primality Testing: Fermats and Eulers theorem, Chinese Reminder theorem, Discrete Logarithms.		
Module:2	**Symmetric Encryption Algorithms**	**7 hours**
Symmetric key cryptographic techniques: Introduction to Stream cipher, Block cipher: DES, AES,IDEA, Block Cipher Operation, Random Bit Generation and RC4		
Module:3	**Asymmetric Encryption Algorithm and Key Exchange**	**8 hours**
Asymmetric key cryptographic techniques: principles, RSA, ElGamal, Elliptic Curve cryptography, Homomorphic Encryption and Secret Sharing, Key distribution and Key exchange protocols, Diffie-Hellman Key Exchange, Man-in-the-Meddle Attack		
Module:4	**Message Digest and Hash Functions**	**5 hours**
Requirements for Hash Functions, Security of Hash Functions, Message Digest (MD5), Secure Hash Function (SHA),Birthday Attack, HMAC		
Module:5	**Digital Signature and Authentication Protocols**	**7 hours**
Authentication Requirements, Authentication Functions, Message Authentication Codes, Digital Signature Authentication, Authentication Protocols, Digital Signature Standards, RSA Digital Signature, Elgamal based Digital Signature, Authentication Applications: Kerberos, X.509 Authentication Service, Public Key Infrastructure (PKI)		
Module:6	**Transport Layer Security and IP Security**	**4 hours**
Transport-Layer Security, Secure Socket Layer(SSL),TLS, IP Security: Overview: IP Security Architecture, Encapsulating Payload Security		
Module:7	**E-mail, Web and System Security**	**7 hours**
Electronic Mail Security, Pretty Good Privacy (PGP), S/MIME, Web Security: Web Security Considerations, Secure Electronic Transaction Protocol Intruders, Intrusion Detection, Password Management, Firewalls: Firewall Design Principles, Trusted Systems.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. Cryptography and Network Security-Principles and Practice, 8th Edition, by Stallings

	William, published by Pearson, 2020		
Reference Books			
1.	Cryptography and Network Security, 3rd Edition, by Behrouz A Forouzan and Depdeep Mukhopadhyay, published by McGrawHill, 2015		
Mode of Evaluation: CAT, written assignment, Quiz, and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE309P	Cryptography and Network Security Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. Understand various Private and Public Key cryptographic algorithms.
2. To learn about hash functions and digital signature algorithms
3. Acquire knowledge in various network security models

Course Outcome

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

1. Implement various cipher techniques without using standard cryptographic library functions
2. Develop the various hash functions and digital signature algorithms for different applications
3. Develop various secured networking-based application

Indicative Experiments

1.	Consider a sender and receiver who need to exchange data confidentially using symmetric encryption. Write program that implements DES encryption and decryption using a 64 bit key size and 64 bit block size
2.	Consider a sender and receiver who need to exchange data confidentially using symmetric encryption. Write program that implements AES encryption and decryption using a 64/128/256 bits key size and 64 bit block size.
3	Develop an chipper scheme by using RSA
4.	Develop a MD5 hash algorithm that finds the Message Authentication Code (MAC)
5	Find a Message Authentication Code (MAC) for given variable size message by using SHA-128 and SHA-256 Hash algorithm Measure the Time consumptions for varying message size for both SHA-128 and SHA-256.
6	Develop the Digital Siganture standard(DSS)for verifying the legal communicating parties
7	Design a Diffie Hellman multiparty key exchange protocol and perform Man-in-the-Middle Attack.
8	Develop a simple client and server application using SSL socket communication
9	Develop a simple client server model using telnet and capture the packets transmitted with tshark Analyze the pcap file and get the transmitted data (plain text) using any packet capturing library. Implement the above scenario using SSH and observe the data
10	Develop a web application that implements JSON web token
	Total Laboratory Hours 30 hours

Mode of assessment: Continuous Assessment, FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

Discipline Elective

Course code	Course Title	L	T	P	C
BCSE206L	Foundations of Data Science	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To provide fundamental knowledge on data science with querying and analytics required for the field of data science.
2. To understand the process of handling heterogeneous data, pre-process and visualize them for better understanding.
3. To gain the fundamental knowledge on data science tools and gain basic skill set to solve real-time data science problems.

Course Outcome

Upon completion of the course the student will be able to

1. Ability to obtain fundamental knowledge on data science.
2. Demonstrate proficiency in data analytics.
3. Apply advanced tools to work on dimensionality reduction and mathematical operations.
4. Handle various types of data and visualize them using through programming for knowledge representation.
5. Demonstrate numerous open source data science tools to solve real-world problems through industrial case studies.

Module:1	**Data Science Context**	**5 hours**

Need for Data Science – What is Data Science - Data Science Process – Business Intelligence and Data Science – Prerequisites for a Data Scientist – Tools and Skills required.

Module:2	**Databases for Data Science**	**7 hours**

Structured Query Language (SQL): Basic Statistics, Data Munging, Filtering, Joins, Aggregation, Window Functions, Ordered Data, preparing No-SQL: Document Databases, Wide-column Databases and Graphical Databases.

Module:3	**Data Science Methodology**	**8 hours**

Analytics for Data Science – Examples of Data Analytics – Data Analytics Lifecycle: Data Discovery, Data Preparation, Model Planning, Model Building, Communicate Results.

Module:4	**Data Analytics on Text**	**7 hours**

Major Text Mining Areas – Information Retrieval – Data Mining – Natural Language Processing NLP) – Text analytics tasks: Cleaning and Parsing, Searching, Retrieval, Text Mining, Part-of-Speech Tagging, Stemming, Text Analytics Pipeline. NLP: Major components of NLP, stages of NLP, and NLP applications.

Module:5	**Platform for Data Science**	**6 hours**

Python for Data Science –Python Libraries – Data Frame Manipulation with numpy and pandas – Exploration Data Analysis – Time Series Dataset – Clustering with Python – Dimensionality Reduction. Python integrated Development Environments (IDE) for Data Science.

Module:6	**GNU Octave for Mathematical Operations**	**6 hours**

Handling Vectors and Matrices: Multiplication, Transpose, Random Matrix creation, Eigen Vectors and Eigen Values, Determinants. Arithmetic Operations – Set Operations – Plotting Data.

Module:7	**Tableau**	**4 hours**

Tableau Introduction – Dimensions, Measures, Descriptive Statistics, Basic Charts, Dashboard Design Principles, Special Chart Types, Integrate Tableau with Google Sheets.

Module:8	**Contemporary Issues**	**2 hours**

	Total Lecture hours:	**45 hours**
Text Book(s)		
1.	Sanjeev Wagh, Manisha Bhende, Anuradha Thakare, 'Fundamentals of Data Science, CRC Press, 1st Edition, 2022.	
Reference Books		
1.	Avrim Blum, John Hopcroft, Ravindran Kannan, "Foundations of Data Science", Cambridge University Press, First Edition, 2020.	
2.	Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media, 1st Edition, 2015.	
3.	Ani Adhikari and John DeNero, 'Computational and Inferential Thinking: The Foundations of Data Science', GitBook, 2019.	
Mode of Evaluation : Continuous Assessment Tests, Quizzes, Assignment, Final Assessment Test		
Recommended by Board of Studies	12-05-2022	
Approved by Academic Council	No. 66	Date 16-06-2022

Course code	Course Title	L	T	P	C
BCSE207L	Programming for Data Science	2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To provide necessary knowledge on data manipulation and to perform analysis on the practical problems using a programming approach.
2. To generate report and visualize the results in graphical form using programming tools.
3. To learn and implement R programs for data science.

Course Outcome

Upon completion of the course, the student will be able to

1. Engrave and use R language to solve problems.
2. Design a suitable form for analysis from real-time data.
3. Formulate insights from the data through statistical inferences.
4. Evaluate and visualize the results, analyze the performance of the models.

Module:1	**Functions in R**	**2 hours**

Programming with R- Running R Code - Including Comments - Defining Variables, Functions -Built-in R Functions - Loading Functions - Writing Functions - Using Conditional Statements.

Module:2	**Vectors and Lists**	**3 hours**

Vector - Vectorized Operations - Vector Indices - Vector Filtering - Modifying Vectors, Lists - Creating Lists - Accessing List Elements - Modifying Lists- Applying Functions to Lists with lapply().

Module:3	**Data Wrangling**	**4 hours**

Understanding Data - The Data Generation Process - Finding Data - Types of Data - Interpreting Data - Using Data to Answer Questions - Data Frames - Working with Data Frames -Working with CSV Data.

Module:4	**Manipulating Data with dplyr and tidyr**	**5 hours**

Data Manipulation - Core dplyr Functions- Performing Sequential Operations -Analyzing Data Frames by Group - Joining Data Frames Together - dplyr in Action: Analyzing Flight Data- Reshaping Data with tidyr -From Columns to Rows: gather() - From Rows to Columns: spread() - tidyr in Action: Exploring Educational Statistics.

Module:5	**Accessing Databases and Web APIs**	**5 hours**

An Overview of Relational Databases -A Taste of SQL-Accessing a Database from R - Accessing Web APIs -RESTful Requests -Accessing Web APIs from R -Processing JSON Data -APIs in Action: Finding Cuban Food in Seattle.

Module:6	**Data Visualization**	**6 hours**

Designing Data Visualizations - The Purpose of Visualization - Selecting Visual Layouts - Choosing Effective Graphical Encodings - Expressive Data Displays - Enhancing Aesthetics - Creating Visualizations with ggplot2- A Grammar of Graphics - Basic Plotting with ggplot2 - Complex Layouts and Customization - Building Maps- ggplot2 in Action: A case study.

Module:7	**Interactive Visualization in R**	**3 hours**

The Plotly Package - The Rbokeh Package - The Leaflet Package - Interactive Visualization in Action: Exploring Changes to the City of Seattle.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Michael Freeman and Joel Ross, Programming Skills for Data Science: Start Writing

	Code to Wrangle, Analyze, and Visualize Data with R, Addison-Wesley, 2018.		
Reference Books			
1.	Benjamin S. Baumer, Daniel T. Kaplan and Nicholas J. Horton, Modern Data Science with R, Chapman and Hall/CRC, 2021.		
2.	John Mount and Nina Zumel, Practical Data Science with R, 2nd edition, Wiley, 2019.		
Mode of Evaluation : Continuous Assessment Tests, Quizzes, Assignment, Final Assessment Test			
Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE207P	Programming for Data Science Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To provide necessary knowledge on data manipulation and to perform analysis on the practical problems using statistical and machine learning approach.
2. To generate report and visualize the results in graphical form using programming tools.
3. To learn and implement R programs for data science.

Course Outcome

Upon completion of the course, the student will be able to

1. Program and use R language to solve problems.
2. Design a suitable form for analysis from real-time data.
3. Formulate insights from the data through statistical inferences.
4. Evaluate and visualize the results, analyze the performance of the models.

Indicative Experiments		
1.	Functions in R	**4 hours**
2.	Vectors and Lists	**2 hours**
3.	Data Frames	**4 hours**
4.	Handling Missing Data	**4 hours**
5.	Manipulating Data with dplyr and tidyr	**2 hours**
6.	Processing JSON Data	**2 hours**
7.	APIs	**3 hours**
8	Data Visualization	**3 hours**
9.	Interactive Visualization in R	**3 hours**
10.	Case Study	**3 hours**
	Total Laboratory Hours	**30 hours**

Mode of assessment: Continuous assessment / FAT / Oral examination and others

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE208L	Data Mining	2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To introduce the fundamental processes data warehousing and major issues in data mining.
2. To impart the knowledge on various data mining concepts and techniques that can be applied to text mining, web mining etc.
3. To develop the knowledge for application of data mining and social impacts of data mining.

Course Outcome

Upon completion of the course the student will be able to

1. Interpret the contribution of data warehousing and data mining to the decision-support systems.
2. Construct the data needed for data mining using preprocessing techniques.
3. Discover interesting patterns from large amounts of data using Association Rule Mining.
4. Extract useful information from the labeled data using various classifiers and Compile unlabeled data into clusters applying various clustering algorithms.
5. Demonstrate capacity to perform a self-directed piece of practical work that requires the application of data mining techniques.

Module:1	Data Warehousing	4 hours

Introduction to Data warehouse - Data Warehouse models- Data warehouse architecture: Three-tier data warehouse architecture - Data warehouse modeling: Data cube and OLAP – Star and Snowflake Schema.

Module:2	Introduction to Data Mining	3 hours

Introduction to data mining - Data mining functionalities - Steps in data mining process- Classification of data mining systems - Major issues in data mining.

Module:3	Data Preprocessing	3 hours

Data Preprocessing: An overview - Data cleaning - Data integration -Data reduction - Data transformation.

Module:4	Frequent Pattern Mining	4 hours

Frequent Pattern Mining: Basic Concepts and a Road Map - Efficient and scalable frequent item set mining methods: Apriori algorithm, FP-Growth algorithm - Mining frequent item sets using vertical data format.

Module:5	Classification Techniques	5 hours

General approach to classification - Classification by decision tree induction - Bayes classification methods - Model evaluation and selection - Techniques to improve classification accuracy - advanced classification methods: Bayesian belief networks- Lazy learners.

Module:6	Cluster Analysis	5 hours

Types of data in cluster analysis - Partitioning methods - K Medoid Clustering - Density based methods - Grid based methods - Outlier analysis.

Module:7	Data Mining Trends and Research Frontiers	4 hours

Overview of Web mining-Temporal and Spatial mining-Other methodologies of data mining: Statistical data mining- Data mining applications.

Module:8	Contemporary Issues	2 hours

	Total Lecture hours:	**30 hours**
Text Book(s)		
1.	Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, third edition, 2013.	
Reference Books		
1.	Parteek Bhatia, Data Mining and Data Warehousing: Principles and Practical Techniques, Cambridge University Press, 2019.	
2.	Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, Introduction to Data Mining, Pearson, 2nd Edition, 2019.	
Mode of Evaluation : Continuous Assessment Tests, Quizzes, Assignment, Final Assessment Test		
Recommended by Board of Studies	12-05-2022	
Approved by Academic Council	No. 66	Date 16-06-2022

Course code	Course Title	L	T	P	C
BCSE208P	Data Mining Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To introduce the fundamental processes data warehousing and major issues in data mining.
2. To impart the knowledge on various data mining concepts and techniques that can be applied to text mining, web mining etc.
3. To develop the knowledge for application of data mining and social impacts of data mining.

Course Outcome

1. Interpret the contribution of data warehousing and data mining to the decision-support systems.
2. Construct the data needed for data mining using preprocessing techniques.
3. Discover interesting patterns from large amounts of data using Association Rule Mining.
4. Extract useful information from the labeled data using various classifiers and Compile unlabeled data into clusters applying various clustering algorithms.
5. Demonstrate capacity to perform a self-directed piece of practical work that requires the application of data mining techniques.

Indicative Experiments	
1.	Introduction to exploratory data analysis using R.
2.	Demonstrate the Descriptive Statistics for a sample data like mean, median, variance and correlation etc.,
3.	Demonstrate Missing value analysis using sample data.
4.	Demo of Apriori algorithm on various data sets with varying confidence and support.
5.	Demo of FP Growth algorithm on various data sets with varying confidence and support.
6	Demo on Classification Techniques such as Decision Tree (ID3 / CART), Bayesian etc., and using sample data.
7.	Demonstration of Clustering Techniques K-Medoid and Hierarchical.
8.	Demonstration on Document Similarity Techniques and measurements.
9.	Simulation of Page Rank Algorithm.
10.	Demonstration on Hubs and Authorities.
	Total Laboratory Hours **30 hours**

Text Book(s)

Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, third edition, 2013.

Reference Books

Parteek Bhatia, Data Mining and Data Warehousing: Principles and Practical Techniques, Cambridge University Press, 2019.

Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, Introduction to Data Mining, Pearson, 2nd Edition, 2019.

Mode of Assessment: Continuous Assessment / FAT / Oral examination and others

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE209L	Machine Learning	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To teach the theoretical foundations of various learning algorithms.
2. To train the students better understand the context of supervised and unsupervised learning through real-life examples.
3. To understand the need for Reinforcement learning in real – time problems.
4. Apply all learning algorithms over appropriate real-time dataset.
5. Evaluate the algorithms based on corresponding metrics identified.

Course Outcome

At the end of this course, student will be able to:

1. Understand, visualize, analyze and preprocess the data from a real-time source.
2. Apply appropriate algorithm to the data.
3. Analyze the results of algorithm and convert to appropriate information required for the real – time application.
4. Evaluate the performance of various algorithms that could be applied to the data and to suggest most relevant algorithm according to the environment.

Module:1	**Introduction to Machine Learning and Pre-requisites**	**4 hours**
Introduction to Machine Learning – Learning Paradigms – PAC learning – Version Spaces – Role of Machine Learning in Artificial Intelligence applications.		
Module:2	**Supervised Learning – I**	**7 hours**
Linear and Non-Linear examples – Multi–Class & Multi-Label classification – Linear Regression – Multiple Linear Regression – Naïve Bayes Classifier – Decision Trees – ID3 – CART – Error bounds.		
Module:3	**Supervised Learning – II**	**8 hours**
K-NN classifier – Logistic regression – Perceptron – Single layer & Multi-layer – Support Vector Machines – Linear & Non-linear – Metrics & Error Correction.		
Module:4	**Unsupervised Learning**	**9 hours**
Clustering basics (Partitioned, Hierarchical and Density based) - K-Means clustering – K-Mode clustering – Self organizing maps – Expectation maximization – Principal Component Analysis – Kernel PCA – tSNE (t-distributed stochastic neighbor embedding) - Metrics & Error Correction.		
Module:5	**Ensemble Learning**	**5 hours**
Bias – Variance Tradeoff – Bagging and Boosting (Random forests, Adaboost, XG boost inclusive) – Metrics & Error Correction.		
Module:6	**Machine Learning in Practice**	**3 hours**
Class Imbalance – SMOTE – One Class SVM – Optimization of hyper parameters.		
Module:7	**Reinforcement Learning (RL)**	**8 hours**
Basics of RL – RL Framework – Markov Decision Process – Exploration Vs Exploitation - Polices, Value Functions and Bellman Equations – Solution Methods – Q-learning.		
Module:8	**Contemporary Issues**	**1 hour**
	Total Lecture hours:	**45 hours**

Text Book(s)

1.	Ethem Alpaydin, Introduction to Machine Learning, MIT Press, Prentice Hall of India, Third Edition 2014.

2.	Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning series) 2nd edition, A Bradford Book; 2018.			
Reference Books				
1.	Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, MIT Press, 2012.			
2.	Tom Mitchell, Machine Learning, McGraw Hill, 3rd Edition, 1997.			
3.	Charu C. Aggarwal, Data Classification Algorithms and Applications, CRC Press, 2014			
Mode of Evaluation : Continuous Assessment Tests, Quizzes, Assignment, Final Assessment Test				
Recommended by Board of Studies		09-05-2022		
Approved by Academic Council		No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE209P	Machine Learning Lab	0	0	2	1
Pre-requisite	Nil	Syllabus version			
		1.0			

Course Objectives

1. To teach the theoretical foundations of various learning algorithms.
2. To train the students better understand the context of supervised and unsupervised learning through real-life examples.
3. To understand the need for Reinforcement learning in real – time problems.
4. Apply all learning algorithms over appropriate real-time dataset.
5. Evaluate the algorithms based on corresponding metrics identified.

Course Outcome

1. At the end of this course, student will be able to:
2. Understand, visualize, analyze and preprocess the data from a real-time source.
3. Apply appropriate algorithm to the data.
4. Analyze the results of algorithm and convert to appropriate information required for the real – time application.
5. Evaluate the performance of various algorithms that could be applied to the data and to suggest most relevant algorithm according to the environment.

Indicative Experiments

1.	Linear & Multiple Linear Regression
2.	Naïve Bayes classifier
3.	Decision trees – ID3 & CART
4.	Logistic regression
5.	Support Vector Machines – Linear & Non-linear
6.	Single & Multilayer Perceptron
7.	K-NN, K-Means & K-mode clustering
8.	Random – forest
9.	Adaboost, XGboost
10.	Principal component analysis
11.	Self – Organizing maps
12.	Q-Learning
	Total Laboratory Hours **30 hours**

Mode of Evaluation: CAT / Mid-Term Lab/ FAT

Recommended by Board of Studies	09-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE331L	Exploratory Data Analysis	2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. The course introduces the methods for data preparation and data understanding.
2. It covers essential exploratory techniques for understanding multivariate data by summarizing it through statistical and graphical methods.
3. Supports to summarize use of predictive analytics, data science and data visualization.

Course Outcomes

At the end of the course, the student will be able to

1. Handle missing data in the real world data sets by choosing appropriate methods.
2. Summarize the data using basic statistics. Visualize the data using basic graphs and plots.
3. Identify the outliers if any in the data set.
4. Choose appropriate feature selection and dimensionality reduction.
5. Apply Techniques for handling multi-dimensional data.

Module:1	Introduction to Exploratory Data Analysis	4 hours

Introduction to Exploratory Data Analysis (EDA) –Steps in EDA, Data Types: Numerical Data – Discrete data, continuous data – Categorical data – Measurement Scales: Nominal, Ordinal, Interval, Ratio – Comparing EDA with classical and Bayesian Analysis – Software tools for EDA.

Module:2	Data Transformation	4 hours

Transformation Techniques: Performing data deduplication - replacing values – Discretization and binning. Introduction to Missing data, handling missing data: Traditional methods - Maximum Likelihood Estimation.

Module:3	Correlation Analysis and Time Series Analysis	4 hours

Types of analysis: Univariate analysis - bivariate analysis - multivariate analysis. Time Series Analysis (TSA): Fundamentals of TSA - characteristics of TSA – Time based indexing - visualizing time series – grouping time series data - resampling time series data.

Module:4	Data Summarization and Visualization	4 hours

Statistical summary measures, data elaboration, 1-D Statistical data analysis, 2-D Statistical data Analysis, contingency tables, n-D Statistical data analysis. Visualization: Scatter plots – Dot charts - Bar plots.

Module:5	Clustering Algorithms	4 hours

Introduction to Spectral clustering – Document clustering – Minimum Spanning Tree clustering. Overview of Model-based clustering – Expectation-Maximization algorithm – Hierarchical Agglomerative model-based clustering. Outlier detection using Clustering.

Module:6	Dimensionality Reduction	4 hours

Linear Methods: Principal Component Analysis (PCA) – Singular Value Decomposition – Factor Analysis -Intrinsic Dimensionality. Non Linear methods: Multidimensional Scaling – Manifold Learning – Self-Organizing Maps.

Module:7	Model Development and Evaluation	4 hours

Constructing linear regression model – evaluation – computing accuracy – understanding accuracy. Understanding reinforcement learning: Difference between supervised and reinforcement learning – Applications of reinforcement learning.

Module:8	Contemporary Issues	2 hours

	Total Lecture hours:	**30hours**

Text Book(s)	
1.	Suresh Kumar Mukhiya, Usman Ahmed, "Hands-On Exploratory Data Analysis with Python" 1st Edition, 2020, Packt Publishing.
2.	Martinez, W , Martinez A & J.L. Solka : Exploratory Data Analysis with MATLAB, CRC Press, A Chapman & Hall Book, 3rd Edition, 2017
Reference Books	
1.	Michael Jambu, "Exploratory and multivariate data analysis", 1991, 1st Edition, Academic Press Inc.
2.	Charu C. Aggarwal, "Data Mining The Text book", 2015, Springer.
3.	Craig K. Enders, "Applied Missing Data Analysis", 2010, 1st Edition, The Guilford Press.

Mode of Evaluation: CAT / written assignment / Quiz / FAT / Project			
Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE331P	Exploratory Data Analysis Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Emphasize the importance of programming in EDA.
2. Familiarize the student with R programming for various tasks.
3. Explore data structures and file processing facilities in R language.

Course Outcomes

At the end of the course, the student will be able to

1. Engrave simple R programs.
2. Debug and execute R programs using R studio.
3. Implement several algorithms in R language.

	Indicative Experiments	
1.	Data transformation and pre-processing. Write R programs to read data from keyboard and transform it to various ranges like [-3,+3], [-1,+1], [0,1] etc.	**4 hours**
2.	Write R programs to read data from keyboard or text files and compute summary measures like arithmetic mean, median, mode, variance and standard deviation. Also read a set of X,Y values and find covariance and correlation, use statistical techniques to identify outlier data	**6 hours**
3.	Estimation of missing data, global methods, class based methods, multiple imputation methods etc	**6 hours**
4	Exploratory Data Analysis for Structured Data	**4 hours**
4.	Write R programs to implement the k-means clustering algorithm by reading the data and user-specified value of k. Display the characteristics of the clusters found by the algorithm.	**6 hours**
5.	Write R programs for nearest neighbour algorithms for classification	**4 hours**
	Total Laboratory Hours	**30 hours**

Mode of assessment: Continuous assessment / FAT / Oral examination and others

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE332L	Deep Learning	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Introduce major deep neural network frameworks and issues in basic neural networks.
2. To solve real world applications using Deep learning.

Course Outcomes

At the end of this course, student will be able to:

1. Understand the methods and terminologies involved in deep neural network, differentiate the learning methods used in Deep-nets.
2. Identify and apply suitable deep learning approaches for given application.
3. Design and develop custom Deep-nets for human intuitive applications.
4. Design of test procedures to assess the efficiency of the developed model.
5. To understand the need for Reinforcement learning in real – time problems.

Module:1	Introduction to neural networks and deep neural networks	7 hours

Neural Networks Basics - Functions in Neural networks – Activation function, Loss function - Function approximation - Classification and Clustering problems - Deep networks basics - Shallow neural networks – Activation Functions – Gradient Descent – Back Propagation – Deep Neural Networks – Forward and Back Propagation – Parameters – Hyperparameters.

Module:2	Improving deep neural networks	8 hours

Mini-batch Gradient Descent – Exponential Weighted Averages – Gradient Descent with Momentum – RMSProp and Adam Optimization – Hyperparameter tuning – Batch Normalization – Softmax Regression – Softmax classifier – Deep Learning Frameworks – Data Augmentation - Under-fitting Vs Over-fitting.

Module:3	Convolution neural networks	6 hours

Foundations of Convolutional Neural Networks – CNN operations – Architecture – Simple Convolution Network – Deep Convolutional Models – ResNet, AlexNet, InceptionNet and others.

Module:4	Recurrent networks	6 hours

Recurrent Neural Networks - Bidirectional RNNs, Encoder, Decoder, Sequence-to-Sequence Architectures, Deep Recurrent Networks, Auto encoders - Bidirectional Encoder Representations from Transformers (BERT).

Module:5	Recursive neural networks	6 hours

Long-Term Dependencies - Echo State Networks - Long Short-Term Memory and Other Gated RNNs - Optimization for Long-Term Dependencies - Explicit Memory.

Module:6	Advanced Neural networks	6 hours

Transfer Learning – Transfer Learning Models – Generative Adversarial Network and their variants – Region based CNN – Fast RCNN - You Only Look Once – Single shot detector.

Module:7	Deep reinforcement learning	5 hours

Deep Reinforcement Learning – Q-Learning – Deep Q-Learning – Policy Gradients - Advantage Actor Critic (A2C) and Asynchronous Advantage Actor Critic (A3C) – Model based Reinforcement Learning – Challenges.

Module:8	Contemporary issues	1 hour
	Total Lecture hours:	45 Hours

Text Book(s)

1.	Ian Goodfellow Yoshua Bengio Aaron Courville, Deep Learning, MIT Press, 2017.			
2	Michael Nielsen, Neural Networks and Deep Learning, Determination Press, first Edition, 2013.			
Reference Books				
1.	N D Lewis, Deep Learning Step by Step with Python, 2016.			
2.	Josh Patterson, Adam Gibson, Deep Learning: A Practitioner's Approach, O'Reilly Media, 2017.			
3	Umberto Michelucci, Applied Deep Learning. A Case-based Approach to Understanding Deep Neural Networks, Apress, 2018.			
4	Giancarlo Zaccone, Md. RezaulKarim, Ahmed Menshawy, Deep Learning with TensorFlow: Explore neural networks with Python, Packt Publisher, 2017.			
Mode of Evaluation: CAT / Written Assignment / Quiz / FAT				
Recommended by Board of Studies		09-05-2022		
Approved by Academic Council		No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE332P	Deep Learning Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Introduce major deep neural network frameworks and issues in basic neural networks.
2. To solve real world applications using Deep learning.

Course Outcomes

At the end of this course, student will be able to:

1. Understand the methods and terminologies involved in deep neural network, differentiate the learning methods used in Deep-nets.
2. Identify and apply suitable deep learning approaches for given application.
3. Design and develop custom Deep-nets for human intuitive applications.
4. Design of test procedures to assess the efficiency of the developed model.
5. Understand the need for Reinforcement learning in real – time problems.

Indicative Experiments

1.	Demonstration and implementation of Shallow architecture, using Python, Tensorflow and Keras. • Google Colaboratory - Cloning GitHub repository, Upload Data, Importing Kaggle's dataset, Basic File operations • Implementing Perceptron, • Digit Classification : Neural network to classify MNIST dataset	**10 hours**
2.	Hyper parameter tuning and regularization practice - • Multilayer Perceptron (BPN) • Mini-batch gradient descent,	**4 hours**
3.	Convolution Neural Network application using Tensorflow and Keras, • Classification of MNIST Dataset using CNN • Face recognition using CNN	**4 hours**
4.	Object detection using Transfer Learning of CNN architectures	**2 hours**
5.	Image denoising (Fashion dataset) using Auto Encoders • Handling Color Image in Neural Network aka Stacked Auto Encoders (Denoising)	**2 hours**
6.	Text processing, Language Modeling using RNN	**2 hours**
7.	Transfer Learning models for classification problems	**2 hours**
8.	Sentiment Analysis using LSTM	**2 hours**
9.	Image generation using GAN	**2 hours**
	Total Laboratory Hours	**30 hours**

Mode of Evaluation: CAT / Mid-Term Lab/ FAT

Recommended by Board of Studies	09-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE333L	Statistical Inference	2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To study statistical methods for hypotheses testing and solving inference problems.
2. To interpret the results in a way that draws evidence-based and well-informed decisions from data.
3. To derive conclusions from data and analyze its implications.

Course Outcomes

At the end of the course, the student will be able to

1. Understand the notion of a parametric model, point estimation of the parameters and properties of a good estimator.
2. Learn the concept of interval estimation and confidence intervals.
3. Understand and perform large-sample tests of hypotheses.
4. Discuss nonparametric tests of hypotheses.
5. Translate and correlate the statistical analysis into Statistical inference

Module:1	Introduction to Estimator	4 hours

Population, sample, parameter and statistic- Estimator, Estimate-characteristics of a good estimator – Unbiasedness- Consistency-Invariance property of Consistent estimator- Sufficient condition for consistency- Sufficiency- Factorization Theorem- Minimal sufficiency- Efficiency- Applications of Lehmann-Scheffe's theorem, Rao - Blackwell Theorem and applications. Bayesian Estimation.

Module:2	Point Estimation	5 hours

Methods of point estimation- Maximum likelihood method (the asymptotic properties of ML estimators are not included), Large sample properties of ML estimator (without proof)- applications of MLE, Method of Minimum variance, method of moments, method of least squares, method of minimum chi-square.

Module:3	Interval Estimation	3 hours

Confidence limits and confidence coefficient; Duality between acceptance region of a test and a confidence interval; Construction of confidence intervals for population proportion (small and large samples) and between two population proportions(large samples); Confidence intervals for mean and variance of a normal population; Difference between the mean and ratio of two normal populations.

Module:4	Testing of hypotheses	4 hours

Types of errors, power of a test, most powerful tests; Neyman-Pearson Fundamental Lemmaand its applications; Notion of Uniformly most powerful tests; Likelihood Ratio tests: Description and property of LR tests - Application to standard distributions.

Module:5	Large sample tests	4 hours

Large sample properties; Tests of significance (under normality assumption)- Test for a single population mean, proportion; Test for equality of two means, proportions; Test for variance, Test for correlation and Test for Regression.

Module:6	Small sample tests	4 hours

Student's t-test, test for a population mean, equality of two population means, paired t-test, F-test for equality of two population variances; Chi-square test for goodness of fit, independence of attributes.

Module:7	Non-parametric tests	4 hours

Sign test, Wilcoxon Signed rank test, Median test, Wilcoxon-Mann-Whitney test, Run test and One sample Kolmogorov Smirnov test, Kruskal Wallis-H-test: Description, properties and applications.

Module:8	**Contemporary Issues**	**2 hours**
	Total hours	**30 hours**

Text Book(s)	
1.	Robert V Hogg, Elliot A Tannis and Dale L.Zimmerman, Probability and Statistical Inference, 9th Edition, Pearson publishers, 2015.
2.	Manoj Kumar Srivastava and Namita Srivastava, Statistical Inference Testing of Hypotheses, Prentice Hall of India, Kindle Edition, 2014.
Reference Books	
1.	Marc S. Paolella, Fundamental statistical inference: A computational approach, Wiley, 2018.
2.	B. K. Kale and K. Muralidharan, Parametric Inference, Narosa Publishing House, 2016.
3.	Miller, I and Miller, M, John E. Freund's Mathematical statistics with Applications, Pearson Education, 2002.
4.	George Casella and Roger L.Berger, Statistical Inference, 2nd edition, Casebound Engelska, 2002.

Mode of Evaluation: CAT / written assignment / Quiz / FAT / Project / Seminar			
Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE333P	Statistical Inference Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To study statistical methods for hypotheses testing and solving inference problems.
2. To interpret the results in a way that draws evidence-based and well-informed decisions from data.
3. To derive conclusions from data and analyze its implications**.**

Course Outcomes

At the end of the course, the student will be able to

1. Understand the notion of a parametric model, point estimation of the parameters and properties of a good estimator.
2. Conquer the concept of interval estimation and confidence intervals.
3. Analyze and perform large-sample tests of hypotheses.
4. Discuss nonparametric tests of hypotheses.
5. Translate and correlate the statistical analysis into Statistical inference

Indicative Experiments		
1	Methods of Estimation – MLE and Method of Moments	**2 hours**
2	Estimation of Confidence intervals	**4 hours**
3	*P*- value and Power of the test	**2 hours**
4	Large Sample Tests- Test for Population mean & Population proportions	**4 hours**
5	Small Sample Tests – t – test for population mean, Paired t-test	**4 hours**
6	F- test for population variances	**2 hour**
7	Chi-square test for goodness of fit and test for attributes	**4 hours**
8	Test for correlation and test for regression	**6 hours**
9	Non-parametric tests	**4 hours**
	Total Laboratory Hours	**30 hours**

Mode of assessment: Continuous assessment / FAT / Oral examination and others

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course Code	Course Title	L	T	P	C
BCSE334L	Predictive Analytics	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Learn the fundamental principles of analytics for business and learn how to Visualize and explore data to better understand relationships among variables.
2. To understand the techniques of modeling and examine how predictive analytics can be used in decision making.
3. Apply predictive models to generate predictions for new data.

Expected Course Outcome

Upon completion of the course the student will be able to

1. Understand the importance of predictive analytics and processing of data for analysis.
2. Describe different types of predictive models.
3. Apply regression and classification model on applications for decision making and evaluate the performance.
4. Analyze the impact of class imbalance on performance measure for model predictions and models that can mitigate the issue during training.
5. Define and apply time series forecasting models in a variety of business contexts.

Module:1	Introduction to Analytics	5 hours

Introduction to predictive analytics – Business analytics: types, applications- Models: predictive models – descriptive models – decision models - applications - analytical techniques.

Module:2	Data Pre-processing and Model Tuning	6 hours

Data transformations: Individual predictors, Multiple predictors, Dealing with missing values, Removing. Adding, Binning Predictors, Computing, Model Tuning, Data Splitting, Resampling.

Module:3	Predictive Modeling	6 hours

Propensity models, cluster models, collaborative filtering, applications and fundamental limitations. Statistical Modeling- Formal Definition, Model Comparison, Classification.

Module:4	Comparison of Regression Models	7 hours

Measuring Performance in Regression Models - Linear Regression and Its Cousins - Non-Linear Regression Models - Regression Trees and Rule-Based Models Case Study: Compressive Strength of Concrete Mixtures.

Module:5	Comparison of Classification Models	7 hours

Measuring Performance in Classification Models - Discriminant Analysis and Other Linear Classification Models - Non-Linear Classification Models - Classification Trees and Rule-Based Models - Model Evaluation Techniques.

Module:6	Remedies for Severe Class Imbalance	6 hours

The Effect of Class Imbalance - Model Tuning - Alternate Cutoffs - Adjusting Prior Probabilities - Unequal Case Weights - Sampling Methods - Cost-Sensitive Training. Measuring Predictor Importance - Factors that can affect Model Performance.

Module:7	Time Series Analysis	6 hours

Methods for time series analyses – Analysis: Motivation – Exploratory analysis – Prediction and forecasting – Classification – Regression analysis – Signal estimation – Segmentation. Models – Autoregressive model - Partial autocorrelation function.

Module:8	Contemporary Issues	2 hours
	Total Lecture Hours:	45 hours

Text Book(s)			
1.	Kuhn, Max, and Kjell Johnson. Applied Predictive Modeling, 3rd Edition, Springer, 2019.		
2.	Jeffrey Strickland, Predictive analytics using R, Simulation educators, Colorado Springs, 2015.		
Reference Books			
1.	Anasse Bari, Mohamed Chaouchi, Tommy Jung, Predictive Analytics for dummies, 2nd edition Wiley, 2016.		
2.	Daniel T.Larose and Chantal D.Larose, Data Mining and Predictive Analytics, 2nd edition Wiley, 2015.		
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Project / Seminar			
Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE335L	Healthcare Data Analytics	3	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Describe how data-based healthcare can help in improving outcomes for patient health.
2. To design data models that combine patient records from multiple sources to form a patient centric view of data.
3. To use data analytics to find health concerns and solutions to the problem faced by a patient.
4. To find meaningful patterns and trends in healthcare data to help the overall population.

Course Outcomes

At the end of the course, the student will be able to

1. Explain the concepts of Healthcare Data Analytics and healthcare foundations.
2. Apply machine learning techniques on healthcare data analytics.
3. Measure and analyse the quality of health-care systems.
4. Develop models for effective predictions in healthcare applications.
5. Use modern day emerging technologies in healthcare data analytics process.

Module:1	Introduction to Healthcare Data Analytics	3 hours

Introduction – Need for Healthcare Analytics - Foundations of Healthcare Analytics – Examples of Healthcare Analytics.

Module:2	Healthcare Foundations	5 hours

Healthcare delivery - Healthcare financing - Healthcare policy – Handling Patient data: the journey from patient to computer - Standardized clinical codesets - Breaking down healthcare analytics: population, medical task, data format, disease.

Module:3	Machine Learning Foundations for Healthcare	8 hours

Model frameworks for medical decision making: Tree-like reasoning, Probabilistic reasoning and Bayes theorem, Criterion tables and the weighted sum approach, Pattern association and neural networks - Machine learning pipeline: Loading the data, Cleaning and preprocessing the data, Exploring and visualizing the data, Selecting features, Training the model parameters, Evaluating model performance.

Module:4	Measuring Healthcare Quality	8 hours

Introduction to healthcare measures, Medicare value-based programs: The Hospital Value-Based Purchasing (HVBP) program, The Hospital Readmission Reduction (HRR) program, The Hospital-Acquired Conditions (HAC) program, The End-Stage Renal Disease (ESRD) quality incentive program, The Skilled Nursing Facility Value-Based Program (SNFVBP), The Home Health Value-Based Program (HHVBP), The Merit-Based Incentive Payment System (MIPS).

Module:5	Making Predictive Models in Healthcare	8 hours

Introduction to Predictive Analytics – Obtaining and Importing the NHAMCS Dataset – Making the Response Variable - Splitting the Data into Train and Test Sets - Preprocessing the Predictor Variables – Building the Models – Using the Models to Make Predictions – Improving our Models.

Module:6	Healthcare Analytics Applications	6 hours

Introduction - Descriptive Analytics Applications - Predictive Analytics Applications - Prescriptive Analytics Application.

Module:7	Healthcare and Emerging Technologies	5 hours

Healthcare analytics and the internet - Healthcare and the Internet of Things - Healthcare

analytics and social media - Healthcare and deep learning - Obstacles, ethical issues, and limitations.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours	**45 hours**

Text Book(s)	
1.	Kumar, Vikas Vik. Healthcare Analytics Made Simple: Techniques in healthcare computing using machine learning and Python. Packt Publishing Ltd, 2018.
2.	El Morr, Christo, and Hossam Ali-Hassan. Analytics in healthcare: a practical introduction. Springer, 2019.
Reference Books	
1.	Dinov, Ivo D. "Data Science and Predictive Analytics." Springer, Ann Arbor, MI, USA https://doi. org/10 1007 (2018): 978-3.
2.	Yang, Hui, and Eva K. Lee, eds. Healthcare analytics: from data to knowledge to healthcare improvement. John Wiley & Sons, 2016.
Mode of Evaluation: CAT / written assignment / Quiz / FAT / Project / Seminar / group discussion	

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE336L	Financial Data Analytics	2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To learn to model financial time series using liner ARMA type time series.
2. To study and analyze to test and model heteroscedastic effects using ARCH / GARCH type time series.
3. To learn how to test for unit root and construct ARMA models.

Course Outcomes

At the end of the course, the student will be able to

1. Approach and analyze any financial data.
2. Differentiate between various time series models.
3. Perform cross-validation of various financial models developed.
4. Forecast future observations on financial data.

Module:1	**Financial data and their properties**	**4 hours**

Asset Returns – Bond Yields and Prices – Implied Volatility – Examples and Visualization of financial data – Multivariate returns.

Module:2	**Linear models for financial time series**	**4 hours**

Simple autoregressive models – Simple moving average models – Simple ARMA models – Unit Root nonstationarity – Exponential smoothing.

Module:3	**Seasonal and Long memory models**	**4 hours**

Seasonal models – Regression models with time series errors – Long memory models.

Module:4	**Asset Volatility and Volatility models**	**4 hours**

Characteristics of Volatility – Structure of a model – Testing for ARCH Effect – ARCH Model – GARCH Model – GARCH-M Model – Exponential Garch Model – Threshold GARCH model – Stochastic volatility model – alternative approaches.

Module:5	**Applications of Volatility Models**	**4 hours**

Garch Volatility Term structure – Option pricing and hedging – Time Varying Correlations and Betas – Minimum Variance Portfolios – Prediction.

Module:6	**High Frequency Financial Data**	**4 hours**

Nonsynchronous trading – Bid ask spread of trading prices – Empirical characteristics of trading data – Models for price changes.

Module:7	**Value at Risk**	**4 hours**

Risk measure and Coherence – Risk metrics –Extreme value approach to Value at Risk – Peak over thresholds.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)

1. Ruey S. Tsay An Introduction to Analysis of Financial Data with R, Wiley, 2013.

Reference Books

1. Analysis of Financial Time Series, by Ruey S. Tsay, 3rd edition, Wiley Series in Probability and Statistics, 2010.
2. William G. Foote, Financial Engineering Analytics: A Practice Manual Using R, 2018.
3. Statistical Analysis of Time-Series Data in SPlus, by Ren´e Carmona, Springer, March 4, 2004.

Mode of Evaluation: CAT / written assignment / Quiz / FAT / Project / Seminar

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

Course code	Course Title	L	T	P	C
BCSE336P	Financial Data Analytics Lab	0	0	2	1
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. Learn how to model financial time series using liner ARMA type time series.
2. Study how to test and model heteroscedastic effects using ARCH / GARCH type time series.
3. Acquire how to test for unit root and construct ARMA models.

Course Outcome

At the end of the course, the student will be able to

1. Approach and analyze any financial data.
2. Differentiate between various time series models.
3. Perform cross-validation of various financial models developed.
4. Forecast future observations on financial data.

Indicative Experiments		
1.	Given a simple daily return of a concern as data, implement and execute a R program to compute the sample mean, standard deviation, skewness, excess kurtosis, minimum and maximum of each simple return series.	**8 hours**
2.	Consider the daily range (daily high–daily low) of Apple stock from January 2, 2007 to December 23, 2011. One can obtain the data by the package quantmod from Yahoo. Compute the first 100 lags of ACF of the series. Is there evidence of long-range dependence? Why? If the range series has long memory, build an ARMA model for the data.	**8 hours**
3.	Consider the 30-year conventional mortgage rates from April 1971 to November 2011. Build a pure time series model for the monthly mortgage rate. Perform model checking and find the fitted model.	**8 hours**
4.	Use the quantmod package to obtain the daily prices of Apple stock from January 2, 2007, to November 30, 2011. Use an ARMA–GARCH model to obtain the daily volatility of the stock. Compare the three volatility series.	**6 hours**
	Total Laboratory Hours	**30 hours**

Mode of assessment: Continuous assessment / FAT / Oral examination and others

Recommended by Board of Studies	12-05-2022		
Approved by Academic Council	No. 66	Date	16-06-2022

BCSE310L	IoT Architectures and Protocols	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart knowledge on the infrastructure, sensor technologies and networking technologies of Internet of Things.
2. To analyze, design and develop solutions for Internet of Things.
3. To explore the real-life aspects of Internet of Things.

Course Outcomes

At the end of this course, student will be able to:

1. Identify the hardware and software components, challenges of Internet of Things.
2. Assess different Internet of Things technologies and their applications.
3. Design basic circuits using sensors interfacing, data conversion process and shield libraries to interface with the real world.
4. Build and demonstrate the project successfully by sensor requirements, coding, emulating and testing.

Module:1	**IoT Fundamentals**	**5 hours**

Definition and Characteristics of Internet of Things (IoT) - Challenges and Issues - Physical Design of IoT - Logical Design of IoT - IoT Functional Blocks.

Module:2	**IoT Communication Architectures and Protocols**	**7 hours**

Control Units – Communication modules – Bluetooth – Zigbee – WiFi – GPS - IoT Protocols (IPv6, 6LoWPAN, RPL, CoAP) – MQTT - Wired Communication - Power Sources.

Module:3	**Technologies Behind IoT**	**5 hours**

Four pillars of IoT paradigm: RFID, Wireless Sensor Networks, Supervisory Control and Data Acquisition (SCADA) - M2M - IoT Enabling Technologies: BigData Analytics, Cloud Computing, Embedded Systems.

Module:4	**Programming the Microcontroller for IoT**	**5 hours**

Working principles of sensors – IoT deployment for Raspberry Pi /Arduino/Equivalent platform – Reading from Sensors, Communication: Connecting microcontroller with mobile devices - Communication through Bluetooth - WiFi and USB - Contiki OS - Cooja Simulator.

Module:5	**Resource Management in IoT**	**5 hours**

Scalability: Network Configuration Protocol, Open vSwitch Database Management Protocol - Routing and Protocols: Collection Tree, LOADng.

Module:6	**IoT to Web of Things**	**9 hours**

Scope of Web of Things (WoT) – IoT Data Management: Set up cloud environment, Cloud access from sensors, Data Analytics Platforms for IOT- Resource Identification: Richardson Maturity Model - REST API.

Module:7	**Applications of IoT**	**7 hours**

Business models for IoT - Green energy buildings and infrastructure - Smart farming - Smart retailing and Smart fleet management

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)			
1.	Simone Cirani, Gianluigi Ferrari, Marco Picone, Luca Veltri. Internet of Things: Architectures, Protocols and Standards, 2019, 1st Edition, Wiley Publications, USA.		
Reference Books			
1.	Bahga, Arshdeep, and Vijay Madisetti. Internet of Things: A Hands-on Approach, 2014,1st Edition, Universities press, India.		
2.	Vlasios Tsiatsis, Jan Holler, Catherine Mulligan, Stamatis Karnourskos and David Boyle. Internet of Things: Technologies and Applications for a New Age of Intelligence, 2018, 2nd Edition, Academic Press, USA.		
Mode of Evaluation: CAT, Written Assignment, Quiz, FAT, Project			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE311L	Sensors and Actuator Devices	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To create a conceptual understanding of the basic principles of sensors, actuators, and their operations
2. To analyze the real-world problems and provide solutions using sensors and actuators
3. To promote awareness regarding recent developments in the fields of sensors and actuators

Course Outcomes

At the end of this course, student will be able to:

1. Classify different Sensors & Actuators based on various physical phenomena and differentiate their performance characteristics
2. Analyze the working principles of thermal, optical & electric sensors and actuators to interpret their mathematical model
3. Interpret the functional principles of magnetic, thermal & Chemical sensors and actuators to interpret their mathematical model
4. Select the relevant sensors and actuators to design real-time data acquisition from ambience via case studies

Module:1	**Overview of Sensors and Actuators**	**4 hours**

The five senses: vision, hearing, smell, taste, and touch – Definitions: Sensors & Actuators – Overview of Sensor and Actuator classifications – Performance characteristics of Sensors & Actuators: Transfer Function, Range, Span, Input and Output Full Scale, Resolution, and Dynamic Range - Calibration & Reliability

Module:2	**Temperature Sensors and Thermal Actuators**	**3 hours**

Thermoresistive sensors: Thermistors, Resistance temperature, and silicon resistive sensors – Thermoelectric sensors – Other Temperature sensors: Optical and Acoustical – Thermomechanical Sensors and Actuators – Case study: B*reath analyzer* using temperature

Module:3	**Optical Sensors and Actuators**	**4 hours**

Principles of Optics: Optical units – Quantum effects – Quantum-based Optical sensors – Photoelectric sensors – Charge coupled device (CCD) based – Thermal-based Optical sensors – Active infrared (AFIR) sensors – Optical Actuators – Case study: Liquid Level Indicator using Optical Sensor**s**

Module:4	**Electric and Magnetic Sensors and Actuators**	**4 hours**

Principles of Electric and Magnetic fields: Basic units – The Electric field: Capacitive Sensors & Actuators – Magnetic sensors and actuators – Magnetoresistance – Magnetostrictive Sensors and Actuators – Magnetometers – Magnetic actuators: Voice Coil Actuators, Motors as Actuators & Magnetic Solenoid Actuators and Magnetic Valves – Case Study: Speed sensing and odometer in a car using smart sensors

Module:5	**Mechanical Sensors and Actuators**	**5 hours**

Definitions and units – Force Sensors: Strain Gauges, Semiconductor Strain Gauges & Tactile Sensors – Accelerometers: Capacitive Accelerometers, Strain Gauge Accelerometers & Magnetic Accelerometers – Pressure Sensors: Mechanical, Piezoresistive, Capacitive & Magnetic – Velocity sensing – Inertial sensors and actuators: Mechanical or Rotor & Optical Gyroscopes – Case study: Tire-pressure monitoring system using smart sensors

Module:6	**Acoustic Sensors and Actuators**	**3 hours**

Definitions and units – Elastic waves and their properties – Microphones: Carbon, Magnetic, Ribbon and Capacitive Microphones – Piezoelectric effect – Piezoelectric Sensors – Acoustic Actuators: Loudspeakers, Headphones and Buzzers - Magnetic and Piezoelectric – Ultrasonic sensors and actuators – Case Study: Ultrasonic parking system		
Module:7	**Chemical Sensors and Actuators**	**5 hours**
Chemical units and Definitions – Electrochemical sensors: Metal Oxide Sensors and Solid Electrolyte Sensors – Potentiometric smart sensors: Glass Membranes, Soluble Inorganic Salt Membrane and Polymer - Immobilized Ionophore Membranes sensors – Thermochemical, Optical, Mass humidity gas sensors – Chemical Actuators: The Catalytic Converter - The Airbag System using smart sensors – Case study: Water quality monitoring system		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 Hours**

Text Book(s)

1.	Nathan Ida, "Sensors, Actuators and their Interfaces - A Multidisciplinary Introduction", 2020, 2nd Edition, IET, United Kingdom.

Reference Books

1.	Jacob Fraden, "Handbook of Modern Sensors Physics, Designs, and Applications", 2016, 5th Edition, Springer, Switzerland.
2.	Subhas Chandra Mukhopadhyay, Octavian Adrian Postolache, Krishanthi P. Jayasundera, Akshya K. Swain, "Sensors for Everyday Life Environmental and Food Engineering", 2017, Volume 23, Springer, Switzerland.

Mode of Evaluation: CAT / Written Assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE311P	Sensors and Actuator Devices Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To create a conceptual understanding of the basic principles of sensors, actuators, and their operations
2. To analyze the real-world problems and provide solutions using sensors and actuators
3. To promote awareness regarding recent developments in the fields of sensors and actuators

Course Outcome

At the end of this course, student will be able to:

1. Classify different Sensors & Actuators based on various physical phenomena and learn various sensor calibration techniques
2. Select the relevant sensors and actuators to design real-time data acquisition from ambience via case studies

Indicative Experiments

1.	Hands-on with the Arduino Programming Environment (IDE) and the different Sensors and Actuators available with the Arduino Kit
2.	Design a data logger with different types of sensors and learn various sensor calibration techniques
3.	Design and implementation of B*reath analyzer* using temperature sensors
4.	Design and implementation of Liquid Level Indicator using optical Sensor**s**
5.	Design and implementation of odometer prototype to sense speed of an automobile
6.	Design and implementation of a prototype to monitor real-time tire-pressure
7.	Develop and validate a prototype for sensing PH and humidity parameters using polymer-based sensors
8.	Design and demonstrate a water quality monitoring system
9.	Demonstrate a simple parking system using ultrasonic sensors
	Total Laboratory Hours **30 hours**

Text Book(s)

1.	Volker Ziemann, "A Hands-On Course in Sensors Using the Arduino and Raspberry Pi", 2018, 1st Edition, CRC Press, United States.

Reference Books

1.	Inamuddin, Rajender Boddula, Abdullah M. Asiri, "Actuators and Their Applications: Fundamentals, Principles, Materials, and Emerging Technologies", 2020, 1st Edition, Wiley-Scrivener, United States.
2.	Peng Zhang, "Industrial Control Technology: A Handbook for Engineers and Researchers", 2008, 1st Edition, William Andrew Inc, United States.

Mode of Evaluation: CAT / Mid-Term Lab/ FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE312L	Programming for IoT Boards	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce Internet of Things (IoT) environment and its technologies for designing smart systems
2. To explore open-source computer hardware/software platform, development and debugging environment, programming constructs and necessary libraries
3. To learn embedded programming constructs and real time systems

Course Outcome

At the end of this course, student will be able to:

1. Investigate various challenges and explore open source hardware prototyping platforms for designing IoT devices
2. Understand basic circuits, sensors and interfacing, data conversion process and shield libraries to interface with the real world
3. Program SBC by exploring protocols, data conversion process, API and expansion boards for practical IoT devices using Python
4. Learn embedded programming constructs and constraints in real time systems for real world socio-economic problems

Module:1	**IoT Ecosystem**	**3 hours**
Challenges and Levels of implementation - Enabling Technologies - Overview of Processing Elements and Peripherals		
Module:2	**Programming for Prototyping Boards**	**4 hours**
Environment: Board, IDE, shields – Programming: syntax, variables, types, operators, constructs and functions – Sketch: skeleton, compile and upload, accessing pins – debugging: UART communication protocol and serial library		
Module:3	**Interfacing for Prototyping Boards**	**5 hours**
Circuits: design, wiring, passive components - sensors and actuators: interfacing, read and write - software libraries – shields - interfacing and libraries		
Module:4	**Programming for Single Board Computers**	**4 hours**
Board schematic – setup - configure and use - OS implications: linux - basics, file system and processes - shell CLI – GUI - Programming API's - RPi.GPIO - PWM library to access pins -Tkinter.		
Module:5	**Interfacing with Single Board Computers**	**5 hours**
Networking - Internet Connectivity - Standard Internet Protocols – MQTT – CoAP - Networking Socket Interface - Cloud - Public APIs and SDK's for accessing cloud services - Social Network APIs - Interfacing - sensors and actuators - Pi Camera - Servo - APIs for data conversion.		
Module:6	**Embedded Programming and RTOS**	**4 hours**
MCU – GPIO – WDT - timers/counters - I/O - A/D - D/A – PWM – Interrupts – Memory - serial communication UART - I2C – SPI - Peripheral Interfacing OS – basics – types – tasks – process - threads (POSIX Threads) - thread preemption - Preemptive Task Scheduling Policies - Priority Inversion - Task communication - Task Synchronization issues - racing and deadlock - binary and counting semaphores (Mutex example) - choosing RTOS		
Module:7	**Real World Projects**	**3 hours**
IoT Integrated Primary Health Care - Face Detection by AI - Cloud IoT Systems for Smart Agriculture - Smart Home Gadgets - Autonomous Car Features – speed and horn intensity control		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)			
1.	Yamanoor, Sai, and Srihari Yamanoor. Python Programming with Raspberry Pi, 2017, 1st edition, Packt Publishing Ltd,. UK		
Reference Books			
1.	Donald Norris, The Internet of Things: Do-It-Yourself Projects with Arduino, Raspberry Pi, and BeagleBone Black, 2015, 1st edition, McGraw Hill Education, India		
2.	Marco Schwartz, Home Automation with Arduino, 3rd edition, Open Home Automation 2014. Schwartz, Marco. Internet of things with arduino cookbook, 2016, 1st edition, Packt Publishing Ltd., UK		
3.	Kooijman, Matthijs. Building Wireless Sensor Networks Using Arduino, 2015, 1st edition, Packt Publishing Ltd., UK		
Mode of Evaluation: CAT / Written Assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE312P	Programming for IoT Boards Lab	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1.To introduce Internet of Things (IoT) environment and its technologies for designing smart systems
2.To explore open-source computer hardware/software platform, development and debugging environment, programming constructs and necessary libraries
3.To learn embedded programming constructs and real time systems

Course Outcome

At the end of this course, student will be able to:
1. Use open-source hardware prototyping platform and peripherals for building digital devices and interactive objects that can sense and control the physical world.
2. Program SBC for practical IoT devices using Python and explore protocols, data conversion process, API's and expansion boards for real world interaction.

Indicative Experiments

1.	Introduction to IoT Development Kit and Development Environment
2.	Internet Controlled LEDs
3.	Temperature Logger
4.	Home Automation
5.	Soil Moisture Sensor
6.	Light Color Control
7.	Home Security System
8.	Parking Sensor
9.	Motor Control
10.	Water Level Control
11.	Street Light Control
	Total Laboratory Hours **30 hours**

Text Book(s)

1.	Yamanoor, Sai, and Srihari Yamanoor. Python Programming with Raspberry Pi, 2017,1st edition, Packt Publishing Ltd,UK.
2.	Donald Norris, The Internet of Things: Do-It-Yourself Projects with Arduino, Raspberry Pi, and BeagleBone Black, 2015,1st edition,McGraw Hill Education, USA.

Reference Books

1.	Schwartz, Marco. Home Automation with Arduino: Automate your Home using Open-Source Hardware. 2013, 1st Edition, CreateSpace Independent Publishing, USA.
2.	Kooijman, Matthijs. Building Wireless Sensor Networks Using Arduino, 2015, 1st edition, Packt Publishing Ltd, UK.

Mode of Evaluation: CAT / Mid-Term Lab/ FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE313L	Fundamentals of Fog and Edge Computing	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce IoT enabling technologies and its opportunities.
2. To review underlying technologies, limitations, and challenges along with performance metrics
   and discuss generic conceptual framework in fog computing.
3. To impart the knowledge to log the sensor data and to perform further data analytics.

Course Outcome

At the end of this course, student will be able to:

1. Explore technologies behind the communication and management of fogs and edge resources.
2. Learn the techniques for storage and computation in fogs, edges, 5G and clouds.
3. Implement Internet of Everything (IoE) applications through fog computing architecture and use optimization techniques for the same.
4. Analyze the performance and issues of the applications developed using fog and edge architecture.

Module:1	**Internet of Things (IoT) and New Computing Paradigms**	**6 Hours**

Introduction - Relevant Technologies - Fog and Edge Computing Completing the Cloud - Hierarchy of Fog and Edge Computing - Business Models – Edge Computing Platforms - Opportunities and Challenges

Module:2	**Challenges in Federating Edge Resources**	**6 Hours**

Introduction - Methodology - Integrated C2F2T Literature by Modeling Technique - Integrated C2F2T Literature by Use - Case Scenarios - Integrated C2F2T Literature by Metrics – Threads - Standards

Module:3	**Orchestration of Network Slices in Fog, Edge, and Clouds**	**6 Hours**

Introduction – Background - Network Slicing - Network Slicing in Software-Defined Clouds- Network Slicing Management in Edge and Fog - Internet of Vehicles (IoV): Architecture, Protocols and Seven-layer security model architecture for Internet of Vehicles - IoV: Network Models, Challenges and future aspects

Module:4	**Optimization Problems in Fog and Edge Computing**	**6 Hours**

Preliminaries - The Case for Optimization in Fog Computing-Formal Modeling Framework for Fog Computing – Metrics - Further Quality Attributes - Optimization Opportunities along the Fog Architecture - Optimization Opportunities along the Service Life Cycle - Toward a Taxonomy of Optimization Problems in Fog Computing

Module:5	**Middleware for Fog and Edge Computing**	**6 Hours**

Need for Fog and Edge Computing Middleware - Design Goals-State-of-the-Art Middleware Infrastructures - System Model - Case Study.

Module:6	**Technologies in Fog Computing**	**7 Hours**

Fog Data Management - Smart Building - Predictive Analysis with FogTorch - Machine Learning in Fog Computing - Data Analytics in the Fog - Data Analytics in the Fog Architecture.

Module:7	**Applications of Fog and Edge Computing**	**6 Hours**

Exploiting Fog Computing in Health Monitoring-Smart Surveillance Video Stream Processing at the Edge for Real - Time Human Objects Tracking-Fog Computing Model for Evolving Smart Transportation Applications - Testing Perspectives of Fog - Based IoT Applications - Legal Aspects of Operating IoT Applications in the Fog

Module:8	**Contemporary Issues**		**2 Hours**
	Total Lecture hours:		**45 Hours**
Text Book(s)			
1.	Buyya, Rajkumar, and Satish Narayana Srirama, Fog and Edge computing: Principles and Paradigms, 2019, 1st edition, John Wiley & Sons, USA.		
Reference Books			
1.	Bahga, Arshdeep, and Vijay Madisetti, Cloud computing: A hands-on approach, 2014, 2nd edition, CreateSpace Independent Publishing Platform, USA.		
2	OvidiuVermesan, Peter Friess, "Internet of Things –From Research and Innovation to Market Deployment", 2014, 1st edition, River Publishers, India.		
Mode of Evaluation: CAT / Digital Assignments/ Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE314L	Privacy and Security in IoT	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart knowledge on the state-of-the-art methodologies and Security in Internet of Things (IoT).
2. To understand the Privacy Preservation and Trust Models in Internet of Things (IoT).
3. To study the Internet of Things (IoT) Security protocols and Security framework.

Course Outcome

At the end of this course, student will be able to:

1. Identify different Internet of Things technologies and their applications.
2. Assess the need for Privacy and security model for the Internet of Things.
3. Explore various Trust Model for IoT and customize real time data for IoT applications.
4. Design security framework and solve IoT security issues.

Module:1	**Security in IoT**	**3 hours**

IoT security: Vulnerabilities, Attacks and Countermeasures - Security Engineering for IoT development - IoT security lifecycle.

Module:2	**Network Robustness and Malware Propagation Control in IoT**	**5 hours**

Network Robustness - Fusion Based Defense Scheme - Sequential Defense Scheme - Location Certificate Based Scheme - Sybil node detection scheme - Formal Modeling and Verification -Sybil Attack Detection in Vehicular Networks - Performance evaluation of various Malware Dynamics Models - Analysis of Attack Vectors on Smart Home Systems.

Module:3	**Blockchain Technology in IoT**	**7 hours**

Technical Aspects - Integrated Platforms for IoT Enablement - Intersections between IoT and Distributed Ledger - Testing at scale of IoT Blockchain Applications - Access Control Framework for Security and Privacy of IoT - Blockchain Applications in Healthcare.

Module:4	**Privacy Preservation in IoT**	**8 hours**

Privacy Preservation Data Dissemination: Network Model, Threat Model - Problem formulation and definition - Baseline data dissemination - Spatial Privacy Graph based data dissemination -Experiment Validation - Smart building concept-Privacy Threats in Smart Building - Privacy Preserving Approaches in Smart Building - Smart Meter Privacy Preserving Approaches.

Module:5	**Privacy Protection in IoT**	**6 hours**

Lightweight and Robust Schemes for Privacy Protection in IoT Applications: One Time Mask Scheme, One Time Permutation Scheme - Mobile Wireless Body Sensor Network - Participatory Sensing

Module:6	**Trust Models for IoT**	**7 hours**

Trust Model Concepts - Public Key Infrastructures Architecture Components - Public Key Certificate Formats - Design Considerations for Digital Certificates - Public Key Reference Infrastructure for the IoT - Authentication in IoT - Computational Security for IoT.

Module:7	**Security Protocols for IoT Access Networks**	**7 hours**

Time Based Secure Key Generation -Security Access Algorithm: Unidirectional, Bidirectional Transmission - Cognitive Security - IoT Security Framework - Secure IoT Layers - Secure Communication Links in IoT - Secure Resource Management, Secure IoT Databases.

Module:8	**Contemporary Issues**	**2 hours**

	Total Lecture hours:		**45 hours**
Text Book(s)			
1.	Hu, Fei. Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations, 2016, 1st edition, CRC Press, USA.		
Reference Books			
1	Russell, Brian and Drew Van Duren. Practical Internet of Things Security, 2016,1st edition, PACKT Publishing Ltd, UK		
2	Kim, S., Deka, G. C., & Zhang, P. (2019). Role of blockchain technology in IoT applications. Academic Press.		
3	Whitehouse O Security of things: An Implementers' guide to cyber-security for internet of things devices and beyond, 2014, 1st edition, NCC Group, UK.		
Mode of Evaluation: CAT, Digital Assignment, Quiz and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE315L	Wearable Computing	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

4. To explore Wearable components and building blocks of Wearable Computing.
5. To enumerate the details of Body Sensor Networks (BSN).
6. To Integrate Wearable and Cloud Computing for BSN applications.

Course Outcomes

At the end of this course, student will be able to:

6. Learn about software, hardware tools, protocols and components required for Wearable Computing.
7. Understand basics of Body Sensor Networks (BSN) and its Programming Framework.
8. Gain Knowledge about Cloud assisted BSN.
9. Learn About the necessary tools required for BSN applications.

Module:1	**Introduction to Wearable Components**	**5 hours**
History - Internet of Things and Wearables - Wearables' Mass Market Enablers - Human Computer Interface and Human Computer Relationship - A Multi Device World.		
Module:2	**Building Blocks for Wearable Computing**	**7 hours**
Bluetooth Low Energy (BLE) - Embedded Software Programming - Sensors for Wearables - Android Wear: Notification Settings and Control, Wear Network - Android Wear API: DataItem – DataMapItem – DataMap - Google Fit API: main package - data sub package		
Module:3	**Body Sensor Networks**	**6 hours**
Typical m-Health System Architecture - Hardware Architecture of a Sensor Node - Communication Medium - Power Consumption Considerations - Communication Standards - Network Topologies - Commercial Sensor Node Platforms - Bio-physiological Signals and Sensors - BSN Application Domains - Developing BSN Applications - Programming Abstractions - Requirements for BSN Frameworks - BSN Programming Frameworks		
Module:4	**Autonomic and Agent-Oriented Body Sensor Networks**	**7 hours**
Task-Oriented Programming in BSNs - SPINE framework - Task-Based Autonomic Architecture - Autonomic Physical Activity Recognition - Agent-Oriented Computing and Wireless Sensor Networks - Mobile Agent Platform for Sun SPOT (MAPS) - Agent-Based Modeling and Implementation of BSNs - Reference Architecture for Collaborative BSNs - C-SPINE: A CBSN Architecture		
Module:5	**Integration of Wearable and Cloud Computing**	**7 hours**
Background - Motivations and Challenges- Reference Architecture for Cloud-Assisted BSNs - BodyCloud: A Cloud-based Platform for Community BSN Applications - Engineering Body Cloud Applications - SPINE Based Design Methodology		
Module:6	**SPINE-Based Body Sensor Network Applications**	**6 hours**
Introduction – Background - Physical Activity Recognition - Step Counter - Emotion Recognition - Handshake Detection - Physical Rehabilitation		
Module:7	**Installing SPINE**	**5 hours**
Introduction - SPINE1.x - Install SPINE 1.x - Use SPINE - Run a Simple Desktop Application Using SPINE1.3 - SPINE Logging Capabilities - SPINE2 - Install SPINE2 - Use the SPINE2 API - Run a Simple Application Using SPINE2		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)				
1.	Fortino, Giancarlo, Raffaele Gravina, and Stefano Galzarano, Wearable computing: from modelling to implementation of wearable systems based on body sensor networks, 2018, 1st edition, John Wiley & Sons, USA			
Reference Books				
1.	Sanjay M. Mishra, Wearable Android™: Android wear & Google Fit app development, 2015, 1st edition, John Wiley & Sons, USA			
2.	Barfield, Woodrow, ed. Fundamentals of wearable computers and augmented reality, 2015, 1st edition, CRC press, USA			
Mode of Evaluation: CAT / Written Assignment / Quiz / FAT				
Recommended by Board of Studies		04-03-2022		
Approved by Academic Council		No. 65	Date	17-03-2022

BCSE316L	Design of Smart Cities	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. To understand the basic concepts of smart cities and their energy sustainability in urban planning.
2. To analyze the security, privacy, and ethics in smart cities planning and development.
3. To perform process control and project management in smart cities.

Course Outcome:

At the end of this course, student will be able to:

1. Ascertain and describe the basic concepts of smart and sustainable cities.
2. Comprehend the knowledge of urban planning and sustainability in smart cities.
3. Analyze the security issues and challenges of smart cities and their advancements.
4. Incorporate project management, planning, and stack holders in the design and development of smart cities.
5. Investigate the various ICT and data analytics to connect government, urban planners, universities, city developers, and communities.

Module:1	**Smart City**	**6 hours**

Smart City - Complexities of Smart Cities - Urban Network - Sensor Network - Role of Urban Networks - Trends in Urban Development - Community Resource Sensing.

Module:2	**Urban Planning**	**6 hours**

Urban Planning - Databases - Principles of Urban Planning - Data Organization - Role of Planning in Smart Cities - Case Studies.

Module:3	**Energy Sustainability in Smart Cities**	**6 hours**

Energy - Decision Making - Energy as a catalyst for Sustainable Transformation - Cohesion and efficiency of smart cities.

Module:4	**Security, Privacy and Ethics in Smart Cities**	**6 hours**

Security challenges in smart cities - Security threats in smart cities - IoT related safety measures for a safer smart city.

Module:5	**Smart Cities Planning and Development**	**6 hours**

City Planning - Understanding Smart Cities - Dimensions of Smart Cities - Global standards and performance benchmark of smart cities - Financing smart cities development - Governance of smart cities.

Module:6	**Process Control and Stabilization**	**7 hours**

Structural concept - Specific applications - Structural health monitoring - Process control and stabilization - Internet of Vehicle (IoV) Importance - Applications - Security issues - Perspectives on Intelligent Transport Systems (ITS) - ITS Highway safety perspective - Environmental aspects of ITS.

Module:7	**Project Management in Smart Cities**	**6 hours**

Case studies on project management of smart cities: web application and mobile based implementation.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)			
1.	Carol L. Stimmel, *Building Smart Cities Analytics, ICT, Design Thinking*, 2016, 1st edition, CRC Press, Taylor and Francis, UK		
Reference Books			
1.	Andrea Vesco and Francesco Ferrero, *Handbook of research on social, economic, and environmental sustainability in the development of smart cities*, 2015, 1st edition, Information Science Reference, IGI Global, USA		
2.	La Scala, Massimo, et al., eds. *From smart grids to smart cities: new challenges in optimizing energy grids*. 2021, Vol. 2. John Wiley & Sons, USA		
3.	Angelakis, Vangelis, et al., eds. *Designing, developing, and facilitating smart cities: urban design to IoT solutions*. 2016, Springer, USA		
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Project / Seminar			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE317L	INFORMATION SECURITY	L	T	P	C
		3	0	0	3
Pre-requisite		**Syllabus version**			
		1.0			

Course Objectives

1. To learn various threats and attacks in a network.
2. To understand and explore fundamental techniques in developing secure applications.
3. To learn various methodologies for securing information systems ranging from operating systems to database management systems and to applications.

Course Outcomes

After completion of this course, the student shall be able to:

1. Apply fundamental knowledge on key security concepts, access control and authentication.
2. Comprehend the use of security techniques for securing the information.
3. Apply various data privacy policies in different areas of web based security systems.
4. Differentiate the needs and application of security in Operating System and Firewalls.
5. Analyze various method of securing databases.

Module:1	**Information Security Concepts**	**4 hours**

Information Security - Computer Security - Threats - Harm - Vulnerabilities - Program Security - Malicious code - Malwares: Viruses, Trojan Horses and Worms - Counter measures.

Module:2	**Authentication and Access Control**	**6 hours**

Authentication - Key management schemes - Hierarchical Key Management Techniques - Security Standards - User Authentication Protocols - Implementing Access Controls - Access Control Models - Role Based Access Control - Attribute Based Access Control - Attribute based Encryption in Information Storage - Physical Access Controls.

Module:3	**Operating Systems Security**	**7 hours**

Security in Operating System - Security in the design of OS: Simplified Design, Layered Design, Kernelized design, Reference Monitor, Trusted Systems, Trusted Systems Functions - Trusted Operating System Design - Rootkit.

Module:4	**Security Countermeasures**	**7 hours**

Design of Firewalls - Types - Personal Firewalls - Configurations - Network Address Translation - Data Loss Prevention - Intrusion Detection and Prevention Systems: Types of IDSs, Intrusion Prevention system, Intrusion Response, Goals of IDSs, Strength and Limitations.

Module:5	**Database Security**	**6 hours**

Database Security - Database Security Requirements - Reliability and Integrity - Sensitive Data - Types of Disclosures - Preventing Disclosures - Inference - Multilevel Databases - Multilevel Security - Database Attacks - SQL Injection Attacks.

Module:6	**Web Security**	**6 hours**

Browser Attacks: Types, Failed Identification and Authentication - Misleading and Malicious Web Contents - Protection against Malicious Web Pages - Website Data: Code within Data, Cross Site Scripting Attacks - Prevention of Data Attacks - Fake e-mails - Spam Detection - Phishing Attacks - Phishing URL Detection and Prevention.

Module:7	**Privacy Issues**	**7 hours**

Privacy Concepts: Aspects of Information Privacy, Computer-Related Privacy Problems - Threats to Personal Data Privacy - People-Based Privacy Concerns - Privacy Principles and Policies - Individual Actions to Protect Privacy - Governments and Privacy - Identify Theft - Privacy issues on the Web Data - Application of Cryptographic Techniques for Privacy Preservation.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book			
1.	Charles P. Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies, Security in Computing, 2018, Fifth Edition, Pearson, New York.		
Reference Books			
1.	Mark Stamp, Information Security: Principles and Practice, 2021, 3rd Edition, Wiley.		
2.	Joanna Lyn Grama, Legal and Privacy Issues in Information Security, 2020, 3rd Edition, Jones and Bartlett Publishers, Inc.		
Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE318L	DATA PRIVACY	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To impart the need of data privacy.
2. To categorize the statistical and computational techniques required to share data, with a primary focus on the social, and health sciences.
3. To formulate architectural, algorithmic, and technological foundations for the maintaining the data privacy.

Course Outcomes

After completion of this course, the student shall be able to:

1. Characterize basic rules, principles for protecting privacy and personally identifiable information.
2. Formulate data that supports useful statistical inference while minimizing the disclosure of sensitive information.
3. Identify the list of threats on the various types of anonymized data.
4. Classify and analyze the methods of test data generation with Privacy and utility.

Module:1	**Data privacy and Importance**	**5 hours**

Need for Sharing Data - Methods of Protecting Data - Importance of Balancing Data Privacy and Utility – Disclosure - Tabular Data - Micro data - Approaches to Statistical disclosure control – Ethics – principles - guidelines and regulations.

Module:2	**Microdata**	**7 hours**

Disclosure - Disclosure risk - Estimating re-identification risk - Non-Perturbative Micro data masking - Perturbative Micro data masking - Information loss in Micro data.

Module:3	**Static Data Anonymization on Multidimensional Data**	**7 hours**

Privacy – Preserving Methods - Classification of Data in a Multidimensional Dataset - Group-based Anonymization: k-Anonymity, l-Diversity, t-Closeness.

Module:4	**Anonymization on Complex Data Structures**	**8 hours**

Privacy-Preserving Graph Data, Privacy-Preserving Time Series Data, Time Series Data Protection Methods, Privacy Preservation of Longitudinal Data, Privacy Preservation of Transaction Data.

Module:5	**Threats to Anonymized Data**	**6 hours**

Threats to Anonymized Data, Threats to Data Structures, Threats by Anonymization Techniques: Randomization, k-Anonymization, l-Diversity, t-Closeness.

Module:6	**Dynamic Data Protection**	**5 hours**

Dynamic Data Protection: Tokenization, Understanding Tokenization, Use Cases for Dynamic Data Protection, Benefits of Tokenization Compared to Other Methods, Components for Tokenization.

Module:7	**Privacy-Preserving Test Data Generation and Privacy Regulations**	**5 hours**

Test Data Fundamentals - Insufficiencies of Anonymized Test Data. Privacy regulations: UK Data Protection Act, Swiss Data Protection Act, HIPPA, General Data Protection Regulation.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. NatarajVenkataramanan, AshwinShriram, Data Privacy: Principles and Practice, 2016, 1st Edition, Taylor & Francis. (ISBN No.: 978-1-49-872104-2), United Kingdom.

Reference Books			
1.	AncoHundepool, Josep Domingo-Ferrer, Luisa Franconi, Sarah Giessing, Eric Schulte Nordholt, Keith Spicer, Peter-Paul de Wolf, Statistical Disclosure Control, 2012, 1st Edition Wiley. (ISBN No.: 978-1-11-997815-2), United States.		
2.	George T. Duncan. Mark Elliot, Juan-Jose Salazar-GonZalez, Statistical Confidentiality: Principle and Practice. 2011, 1st Edition, Springer. (ISBN No.: 978-1-44-197801-1).		
Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE319L	PENETRATION TESTING AND VULNERABILITY ANALYSIS	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the system security-related incidents and insight on potential defenses, countermeasures against common vulnerabilities.
2. To provide the knowledge of installation, configuration, and troubleshooting of information security devices.
3. To make students familiarize themselves with the tools and common processes in information security audits and analysis of compromised systems.

Course Outcome

After completion of this course, the student shall be able to:

1. Familiarized with the basic principles for Information Gathering and Detecting Vulnerabilities in the system.
2. Gain knowledge about the various attacks caused in an application.
3. Acquire knowledge about the tools used for penetration testing.
4. Learn the knowledge into practice for testing the vulnerabilities and identifying threats.
5. Determine the security threats and vulnerabilities in computer networks using penetration testing techniques.

Module:1	**Pentesting Fundamentals**	**5 hours**
Vulnerability Assessment (VA)- Pentesting Analysis (PTA) **-T**ypes of Vulnerability Assessments-Modern Vulnerability Management Program-Ethical Hacking terminology- Five stages of hacking- Vulnerability Research - Impact of hacking - Legal implication of hacking - Compare Vulnerability Assessment (VA) and Penetration Testing (PT) Tools.		
Module:2	**Information Gathering Methodologies**	**5 hours**
Competitive Intelligence- DNS Enumerations- Social Engineering attacks - Scanning and Enumeration. Port Scanning: Network Scanning, Vulnerability Scanning, scanning tools- OS and Fingerprinting Enumeration - System Hacking Password.		
Module:3	**System Hacking**	**3 hours**
Password cracking techniques- Key loggers- Escalating privileges- Hiding Files, Active and Passive sniffing - ARP Poisoning - IP Poisoning and MAC Flooding.		
Module:4	**Wireless Pentesting**	**4 hours**
Wi-Fi Authentication Modes - Bypassing WLAN Authentication - Types of Wireless Encryption - WLAN Encryption Flaws – Access Point Attacks - Attacks on the WLAN Infrastructure - Buffer Overloading.		
Module:5	**The Metasploit Framework**	**3 hours**
Metasploit User Interfaces and Setup - Getting Familiar with MSF Syntax - Database Access - Auxiliary Modules- Payloads - Staged vs Non-Staged Payloads - Meterpreter Payloads - Experimenting with Meterpreter.		
Module:6	**Web Application Attacks**	**4 hours**
Web Application Assessment Methodology – Enumeration - Inspecting URLs - Inspecting Page Content - Viewing Response Headers - Inspecting Sitemaps - Locating Administration Consoles.		
Module:7	**Exploiting Web-Based Vulnerabilities**	**4 hours**
Exploiting Admin Consoles - Cross-Site Scripting (XSS) - SQL Injection.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book(s)			
1.	Najera-Gutierrez G, Ansari JA. Web Penetration Testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux., 2018, 3rd Edition, Packt Publishing Ltd, United Kingdom.		
2.	Hadnagy C. Social engineering: The science of human hacking, 2018, 2nd Edition, John Wiley & Sons, United States.		
Reference Books			
1.	Weidman G. Penetration testing: a hands-on introduction to hacking,2014, 1st Edition, No Starch Press, United States		
2.	Engebretson P. The basics of hacking and penetration testing: ethical hacking and penetration testing made easy, 2013, 2nd Edition, Elsevier.		
Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE319P	PENETRATION TESTING AND VULNERABILITY ANALYSIS LAB	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the system security-related incidents and insight on potential defenses, countermeasures against common vulnerabilities.
2. To provide the knowledge of installation, configuration, and troubleshooting of information security devices.
3. To make students familiarize themselves with the tools and common processes in information security audits and analysis of compromised systems.

Course Outcome

After completion of this course, the student shall be able to:

1. Learn the knowledge into practice for testing the vulnerabilities and identifying threats.
2. Determine the security threats and vulnerabilities in computer networks using penetration testing techniques.

Indicative Experiments	
1.	Perform a track of information about Domain Registrars and DNS by lookup technologies
2.	Perform various Port Scanning methodologies to identify the misconfiguration issues about the infrastructure.
3.	Analyze the traffic routing and information carried among the network through Wireshark
4.	Exploit threats and mitigation strategies for, ARP Spoofing, IP Spoofing,
5.	Demonstrate various approaches followed on password breaking methodology.
6.	Perform and analyze the wireless network to identify their weakness around access points with defensive mechanisms around it.
7.	Apply various payloads to gain various categories of backdoor access of a machine using Metasploit and Meterpreter.
	Total Laboratory Hours **30 hours**

Text Books

1.	Najera-Gutierrez G, Ansari JA. Web Penetration Testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux., 2018, 3rd Edition, Packt Publishing Ltd, United Kingdom.
2.	Hadnagy C. Social engineering: The science of human hacking, 2018, 2nd Edition, John Wiley & Sons, United States.

Reference Books

1.	Weidman G. Penetration testing: a hands-on introduction to hacking,2014, 1st Edition, No Starch Press, United States

Mode of assessment: Continuous assessment / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE320L	WEB APPLICATION SECURITY	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To study and practice fundamental techniques to develop secure web applications.
2. To identify web applications vulnerabilities and understand vulnerability management.
3. To assess web application security attacks and defence.

Course Outcome

After completion of this course, the student shall be able to:

1. Understand security challenges and the need for Authentication and Authorization in web-based systems and applications.
2. Familiarize the Application Programming Interface analysis and vulnerability management of securing a web-based system.
3. Learn the web application hacking techniques and prevention solutions.
4. Apply the best practices of Secure Credentials, session management, and Security Automation in web applications.
5. Develop the best strategies to prevent XSS, CSRF, XXE, Injection, DOS attacks and Securing Third-Party Dependencies.

Module:1	**Web Application Reconnaissance**	**5 hours**

Information Gathering - Web Application Mapping - Structure of Modern Web Application: Modern Versus Legacy Web Applications, REST APIs, JavaScript Object Notation, Browser DOM, SPA Frameworks, Authentication and Authorization Systems, Web Servers, Server-Side Databases, Client-Side Data Stores.

Module:2	**Sub Domain and Application Programming Interface Analysis**	**7 hours**

Sub Domain: Multiple Applications per Domain - Browser's Built-In Network Analysis Tools - Search Engine Caches - Accidental Archives - Social Snapshots - Zone Transfer Attacks - Brute Forcing Subdomains and Dictionary Attacks - Application Programming Interface Analysis(API): Endpoint Discovery and Endpoint Shapes, Authentication Mechanisms.

Module:3	**Web Application Vulnerability**	**6 hours**

Detecting Client-Side and Server-Side Frameworks - Secure Versus Insecure Architecture Signals - Multiple Layers of Security - Adoption and Reinvention - Common Vulnerabilities and Exposures Database

Module:4	**Web Application Hacking**	**6 hours**

Cross-Site Scripting (XSS): XSS Discovery and Exploitation, Stored XSS, Reflected XSS, DOM-Based XSS, Mutation-Based XSS - Cross-Site Request Forgery (CSRF): Query Parameter Tampering, CSRF Against POST Endpoints - XML External Entity (XXE): Direct and Indirect XXE.

Module:5	**Web Application Attacks**	**6 hours**

SQL Injection - Code Injection - Command Injection - Denial of Service (DoS): regex DoS (ReDoS), Logical DoS Vulnerabilities, Distributed DoS - Exploiting Third-Party Dependencies.

Module:6	**Securing Web Applications**	**7 hours**

Defensive Software Architecture - Vulnerability Analysis and Management - Secure Sockets Layer and Transport Layer Security - Secure Credentials, Hash Credentials - Secure-Coding Anti-Patterns - Security Automation: static and dynamic analysis - Vulnerability Regression Testing - Bug Bounty Programs.

Module:7	**Vulnerability Management and Hacking Prevention**	**6 hours**

Common Vulnerability Scoring System - Defending Against attacks: XSS, CSRF, XXE,

Injection, and DOS - Securing Third-Party Dependencies.			
Module:8	**Contemporary Issues**		**2 hours**
	Total Lecture hours:		**45 hours**
Text Book			
1.	Andrew Hoffman, Web Application Security- Exploitation and Countermeasures for Modern Web Applications, March 2020, 1st Edition, O'Reilly Media, California.		
Reference Books			
1.	D. Stuttard and M. Pinto, The Web Applications Hackers Handbook, 2011, 2nd Edition, Indianapolis, IN: Wiley, John Sons, United States.		
2.	Malcolm McDonald, Web Security for Developers: Real Threats, Practical Defense, 2020, Illustrated edition, No Starch Press, United States.		
Mode of Evaluation: CAT, Written Assignment, Quiz, FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE321L	MALWARE ANALYSIS	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the malware taxonomy and malware analysis tools.
2. To identify and analyze malware samples using static, dynamic analysis, and reverse engineering techniques.
3. To detect and analyze malicious documents and mobile malware.

Course Outcome

After completion of this course, the student shall be able to:

1. Possess the skills to carry out static and dynamic malware analysis on various malware samples.
2. Understand the executable formats, Windows internals, and APIs.
3. Apply techniques and concepts to unpack, extract, and decrypt malware.
4. Comprehend reverse-engineering of malware and anti-malware analysis techniques.
5. Achieve proficiency with industry-standard malware analysis tools.

Module:1	**Fundamentals of Malware Analysis**	**5 hours**

Malware taxonomy - Malware analysis techniques – Packed and Obfuscated Malware - Portable Executable File Format: Headers and Sections, Malware Analysis in Virtual Machines - Malware Analysis Tools: ProcMon/ ProcExplore, BinText, FileAlyzer, OllyDbg, etc.

Module:2	**Static Analysis**	**4 hours**

File signature analysis and Identifying file dependencies -Database of file hashes. String analysis - Local and online malware sandboxing - Levels of Abstraction - x86 Architecture - x86/x86_64 Assembly - Static Analysis Tools: PeiD, Dependency Walker, Resource Hacker.

Module:3	**Dynamic Analysis**	**4 hours**

Source level vs. Assembly level Debuggers - Kernel vs. User-Mode Debugging – Exceptions - Modifying Execution with a Debugger - Modifying Program Execution in Practice - DLL analysis - Dynamic Analysis Tools: Virustotal, Malware Sandbox, Windows Sysinternals

Module:4	**Reverse Engineering**	**4 hours**

Reverse engineering malicious code - Identifying malware passwords - Bypassing authentication -Advanced malware analysis: Virus, Trojan and APK Analysis - Reverse Engineering Tools: IDA Pro and OLLYDBG

Module:5	**Malicious Document Analysis**	**3 hours**

PDF and Microsoft Office document structures – Identify PDF and office document vulnerabilities - Analysis of suspicious websites - Examining malicious documents: word, XL, PDF, and RTF files - Malware extraction and analysis tools.

Module:6	**Anti-Reverse-Engineering**	**3 hours**

Anti-Disassembly - Anti-Debugging - Anti-Forensic Malware - Packers and Unpacking – Shellcode Analysis - 64-Bit Malware

Module:7	**Mobile Malware Analysis**	**5 hours**

Mobile application penetration testing - Android and iOS Vulnerabilities - Exploit Prevention - Handheld Exploitation - Android Root Spreading and Distribution Android

Debugging - Machine learning techniques for malware analysis: Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), Decision Trees (DT), Naïve Bayes (NB), and Neural Networks (NN).		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book	
1.	Abhijit Mohanta, Anoop Saldanha, Malware Analysis and Detection Engineering a Comprehensive Approach to Detect and Analyze Modern Malware, 2020, 1st edition, Apress (ISBN 978-1-4842-6192-7), United States.
2.	M. Sikorski and A. Honig, Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software. 2012, 1st edition, No Starch Press San Francisco, CA. (ISBN No.: 9781593272906), United States.
Reference Books	
1.	Monnappa K A, Learning Malware Analysis- Explore the concepts, tools, and techniques to analyze and investigate Windows malware, 2018, 1st edition, Packt Publishing, (ISBN 978-1-78839-250-1), United Kingdom.

Mode of Evaluation: CAT / Assignment / Quiz / FAT / Seminar			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE321P	MALWARE ANALYSIS LAB	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the malware taxonomy and malware analysis tools.
2. To identify and analyze malware samples using static, dynamic analysis, and reverse engineering techniques.
3. To detect and analyze malicious documents and mobile malware.

Course Outcome

After completion of this course, the student shall be able to:

1. Apply techniques and concepts to unpack, extract, and decrypt malware.
2. Achieve proficiency with industry-standard malware analysis tools.

Indicative Experiments	
1	Examining PE Files using PEview, PE explorer and Resource Hacker • Disassembling Portable Executable (PE32) imports, exports, functions, main address, malicious string locations
2	Sandboxing malware using SANDBOX tool, Virus Total Analysis, Anyrun Analysis
3	Basic malware analysis: • file compilation date • imports/ exports, suspicious strings • run-time effect • procmon filter • hist -based signatures revealing files • registry keys, processes, services network-based signatures
4	Advanced static malware analysis • find address of main, code constructs, suspicious strings, • imported functions, their tasks, • intention of the malware impact of the malware via hex code
5	Analyze the malware using IDA Pro for reverse-engineering the malware: strings analysis, local variables, graph mode to cross-references, Analyzing Functions
6	Analyze the malware using OllyDbg: Debug the malware, Viewing Threads and Stacks, OllyDbg Code-Execution Options, Breakpoints, Loading DLLs, Exception Handling
7	Advanced analysis of Windows programs for processes, interactive remote shell, uploaded file, address of the subroutine, return value, Windows APIs
8	Malware behavior analysis • finding the source of malware • persistence mechanism, multiple instances replication mechanisms, hiding strategies • API calls for keylogging, constants involved post-infection actions of the malware, mutex, SendMessage API structure
9	Malware self-defense, packing and unpacking, obfuscation and de-obfuscation using Packers and obfuscation tools
10	Anti-disassembly and anti-debugging techniques used in the binary by patching the PE, set a breakpoint in the malicious subroutine
11	Analyzing malicious Microsoft Office and Adobe PDF documents to locate malicious

	embedded code such as shellcode, VBA macros or JavaScript, disassemble and/ or debug, shellcode analysis			
			Total Laboratory Hours	**30 hours**
Text Book(s)				
1.	M. Sikorski and A. Honig, Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software. 2012, 1st edition, No Starch Press San Francisco, CA. (ISBN No.: 9781593272906), United States.			
Reference Books				
1.	B. Dang, A. Gazet, E. Bachaalany, and S. Josse, Practical Reverse Engineering: X86, X64, arm, Windows Kernel, Reversing Tools, and Obfuscation. , 2014, Wiley, United States. (ISBN No. : 978-1-118-78731-1)			
Mode of assessment: Continuous assessment / FAT				
Recommended by Board of Studies	04-03-2022			
Approved by Academic Council	No.65	Date	17-03-2022	

BCSE322L	DIGITAL FORENSICS	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To present a comprehensive perception of digital forensic principles, collection, preservation, and analysis of digital evidence.
2. To enlighten the importance of forensic procedures, legal considerations, digital evidence controls, and the documentation of forensic analysis.
3. To develop a comprehension of the different tools and methods for conducting digital forensic acquisition and analysis.

Course Outcomes

After completion of this course, the student shall be able to:

1. Understand the responsibilities and liabilities of a computer forensic investigator
2. Seize a computer from a crime scene without damage and follow the legal procedures and standards.
3. Demonstrate the ability to perform forensic data acquisition and analysis.
4. Analyze and retrieve hidden and damaged files from different operating systems.
5. Apply forensics to recent technologies such as smart phones, email, cloud and social media.

Module:1	**Understanding Digital Forensics and Legal Aspects**	**3 hours**

Understanding computer forensics - Preparing for computer investigation – Maintaining professional conduct – understanding computer investigations – Taking a systematic approach – Corporate Hi-Tech investigations – Conducting an investigation.

Module:2	**Acquisition and Storage of Data**	**4 hours**

Understanding Storage Formats for Digital Evidence - Determining the Best Acquisition Method - Contingency Planning for Image Acquisitions - Using Acquisition Tools - Validating Data Acquisitions - Performing RAID Data Acquisitions - Using Remote Network Acquisition Tools - Storing Digital Evidence - Obtaining a Digital Hash - Sample Cases.

Module:3	**Working with Windows**	**5 hours**

Understanding File Systems - Exploring Microsoft File Structures - Examining NTFS Disks - Understanding Whole Disk Encryption - Understanding the Windows Registry - Understanding Microsoft Startup Tasks - Understanding MS-DOS Startup Tasks - Evaluating Computer Forensics Tool Needs - Computer Forensics Software and Hardware Tools.

Module:4	**Working with Linux/Unix Systems**	**4 hours**

UNIX and Linux Overview - Inodes - Boot Process - Drives and Partition Schemes - Examining disk Structures - Understanding Other Disk Structures - Ownership and Permissions, File Attributes, Hidden Files, User Accounts - Case studies - Validating Forensic Data – Addressing Data-Hiding Techniques – Locating and Recovering Graphics File.

Module:5	**Email and Social Media Forensics**	**4 hours**

Investigating E-mail crimes and Violations – Applying Digital Forensics Methods to Social Media Communications - Social Media Forensics on Mobile Devices - Forensics Tools for Social Media Investigations.

Module:6	**Mobile Forensics**	**4 hours**

Mobile phone basics – Acquisition procedures for mobile - Android Device –Android Malware – SIM Forensic Analysis – Case study.

Module:7	**Cloud Forensics**	**4 hours**

Working with the cloud vendor, obtaining evidence, reviewing logs and APIs.			
Module:8	**Contemporary Issues**		**2 hours**
	Total Lecture hours:		**30 hours**
Text Book(s)			
1.	B. Nelson, A. Phillips, F. Enfinger, and C. Steuart, Guide to Computer Forensics and Investigations, 2019, 6th ed. CENGAGE, INDIA (ISBN: 9789353506261)		
Reference Books			
1.	André Årnes, Digital Forensics, 2018, 1st ed., Wiley, USA(ISBN No.: 9781119262411)		
2.	Nihad A Hassan, Digital Forensics Basics: A Practical Guide to Using Windows OS, 2019, 1st ed, APress, USA (ISBN: 9781484238387)		
Mode of Evaluation: CAT, assignment, Quiz and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE322P	DIGITAL FORENSICS LAB	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To present a comprehensive perception of digital forensic principles, collection, preservation, and analysis of digital evidence.
2. To enlighten the importance of forensic procedures, legal considerations, digital evidence controls, and the documentation of forensic analysis.
3. To develop a comprehension of the different tools and methods for conducting digital forensic acquisition and analysis.

Course Outcomes

After completion of this course, the student shall be able to:

1. Demonstrate the ability to perform forensic data acquisition and analysis.
2. Apply forensics to recent technologies such as smart phones, email, cloud and social media.

Indicative Experiments

1.	Extract the features based on various color models and apply on image and video retrieval
2.	File Recovery (Deleted, fragmented, hidden)
3.	Network Forensics (Determining the type attacks, extracting files from network logs, encrypted _les)
4.	OS Forensics (Windows and Linux artifacts, memory, registry)
5.	Mobile Forensics(Tools for Android and iOS)
6.	Mobile Forensics(Tools for Android and iOS)
7.	Social Media Forensics
	Total Laboratory Hours **30 hours**

Text Book

1.	B. Nelson, A. Phillips, F. Enfinger, and C. Steuart, Guide to Computer Forensics and Investigations, 2019, 6th ed. CENGAGE, INDIA (ISBN: 9789353506261)

Reference Books

1.	Nihad A Hassan, Digital Forensics Basics: A Practical Guide to Using Windows OS, 2019, 1st ed, APress, USA (ISBN: 9781484238387)

Mode of assessment: Continuous assessment / FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE323L	DIGITAL WATERMARKING AND STEGANOGRAPHY	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the basic principles, characteristics, various approaches and applications of digital watermarking and steganography.
2. To apply digital watermarking techniques as an authentication tool for distribution of content over the Internet and steganography techniques for covert communication.
3. To impart knowledge on the basics of the counter measures like steganalysis for assessing the data hiding methods.

Course Outcome

After completion of this course, the student shall be able to:

1. Learn the fundamental concepts, principles, characteristics and performance measures of digital watermarking and steganography.
2. Acquire the various concepts of watermarking for digital authentication and authorization schemes related to electronic documents, image and video.
3. Gathering the various concepts of steganography to access the sensitive information concealing of message, image, audio or video within another file.
4. Design and implement efficient data hiding methods against steganalysis techniques.

Module:1	**Fundamentals of Digital Watermarking**	**6 hours**

Importance of Watermarking - Application and Properties of Watermarking - Models of Watermarking - Basic Message Coding: Mapping Message into Message Vectors, Error Correction Coding - Watermarking with Side Information - Analyzing Errors.

Module:2	**Digital Watermarking Schemes**	**7 hours**

Spatial Domain: Correlation based Watermarking, Least Significant bit Watermarking - Frequency domain: Discrete Wavelet Transform Watermarking, Discrete Fourier Transform Watermarking, Discrete Cosine Watermarking, Quantization Watermarking, Haar Transform Watermarking, Hadamard Transform Watermarking - Robust Watermarking - Fragile and Semi Fragile Watermarking.

Module:3	**Digital Watermarking Security and Authentication**	**5 hours**

Watermarking Security: Security Requirements, Watermark Security and Cryptography, Watermarking Attacks and Tools - Content Authentication: Exact Authentication, Selective Authentication, Localization, Restoration.

Module:4	**Steganography**	**7 hours**

Basics and Importance of Steganography - Applications and Properties of Steganography - Steganography: LSB embedding, Steganography in palette images -Steganography in JPEG images: JSteg data hiding in spatial and transform domain -Steganography Security.

Module:5	**Audio and Video Steganography**	**6 hours**

Audio Steganography: Temporal domain techniques, Transform domain techniques, Cepstral Domain - Video Steganography: Introduction Video Streams, Substitution-Based Techniques, Transform Domain Techniques, Adaptive Techniques, Format-Based Techniques - Cover Generation Techniques Video Quality Metrics - Perceptual Transparency Analysis - Robustness against Compression and Manipulation.

Module:6	**Wet Paper Code**	**6 hours**

Random Linear Codes - LT Codes - Perturbed Quantization, Matrix Embedding - Matrix Embedding Theorem - Binary Hamming Codes - Q-Ary Case Random Linear Codes for Large Payloads.

Module:7	**Steganalysis**	**6 hours**

Steganalysis Principles - Statistical Steganalysis: Steganalysis as detection problem,

Modeling images using features, Receiver operating Characteristics - Targeted Steganalysis : Sample pair analysis, Targeted attack on F5 using Calibration, Targeted attack on ± embedding - Blind Steganalysis: Features for steganalysis of JPEG images (cover vs all-stego and one class neighbor machine).		
Module:8	**Contemporary Isues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	Frank Y. Shih, Digital Watermarking and Steganography Fundamentals and Techniques, 2020, 2nd Ed. CRC Press, United States. (ISBN No. : 9780367656430)
2.	J. Fridrich, Steganography in Digital Media: Principles, Algorithms, and Applications, 2010, 1st Ed. Cambridge: Cambridge University Press, United Kingdom. (ISBN No.: 978-0-52-119019-0)
Reference Books	
1.	I. J. Cox, M. L. Miller, J. A. Bloom, T. Kalker, and J. Fridrich, Digital Watermarking and Steganography, 2008, 2nd Ed. Amsterdam: Morgan Kaufmann Publishers In, United States. (ISBN No. : 978-0-12-372585-1)
2.	P. Wayner, Disappearing Cryptography: Information hiding: Steganography and Watermarking, 2008, 3rd ed. Amsterdam: Morgan Kaufmann Publishers In, United States. (ISBN No. : 978-0-08-092270-6)

Mode of Evaluation: CAT / Assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE324L	FOUNDATIONS OF BLOCKCHAIN TECHNOLOGY	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand building blocks of Blockchain.
2. To significance of Distributed Ledger Technology and Smart Contract.
3. To exploit applications of Blockchain in real world scenarios and their impacts.

Course Outcomes

After completion of this course, the student shall be able to:

1. Understand Blockchain ecosystem and its services in real world sceneries
2. Apply and Analyze the requirement of Distributed Ledger Technology and Smart Contract
3. Design and Demonstrate end-to-end decentralized applications
4. Acquaint the protocol and assess their computational requirements

Module:1	**Foundations of Blockchain**	**7 hours**

Blockchain Architecture – Challenges – Applications – Blockchain Design Principles -The Blockchain Ecosystem - The consensus problem - Asynchronous Byzantine Agreement - AAP protocol and its analysis - peer-to-peer network – Abstract Models - GARAY model - RLA Model - Proof of Work (PoW) - Proof of Stake (PoS) based Chains - Hybrid models.

Module:2	**Distributed Ledger Technology**	**6 hours**

Origin of Ledgers – Types and Features of Distributed Ledger Technology (DLT) - Role of Consensus Mechanism - DLT Ecosystem - Distributed Ledger Implementations – Blockchain - Ethereum - Public and Private Ledgers – Registries – Ledgers - Practitioner Perspective: Keyless Technologies, Transparency as a Strategic Risk, Transparency as a Strategic Asset, Usage of Multiple IDs - Zero Knowledge Proofs - Implementation of Public and Private Blockchain

Module:3	**Smart Contracts**	**5 hours**

Anatomy of a Smart Contracts - Life Cycle - Usage Patterns - DLT-based smart contracts - Use Cases: Healthcare Industry and Property Transfer.

Module:4	**Decentralized Organization**	**5 hours**

Decentralization versus Distribution - Centralized-distributed (Ce-Di) organizations - Decentralized-distributed (De-Di) organizations - Decentralized Autonomous Organizations: Aragon, DAOstack, DAOhaus and Colony.

Module:5	**Types of Blockchain Ecosystem**	**7 hours**

One-Leader Ecosystem - Joint Venture or Consortia Ecosystems - Regulatory Blockchain Ecosystems - Components in Blockchain Ecosystem: Leaders, Core Group, Active Participants, Users, Third-Party Service Providers - Governance for Blockchain Ecosystems.

Module:6	**Blockchain Protocols**	**6 hours**

Ethereum tokens – Augur - Golem - Understanding Ethereum tokens - App Coins and Protocol Tokens - Blockchain Token Securities Law Framework - Token Economy - Token sale structure - Ethereum Subreddit.

Module:7	**High Performance Computing**	**7 hours**

Integrity of High Performance Systems - Data Provenance - Cluster Construction and Deployment - Mock Workload - Blockchain Software Evaluation - Blockchain storage of Integrity Data.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. Dhillon, V., Metcalf, D., and Hooper, M, Blockchain enabled applications, 2017, 1st

	Edition, CA: Apress, Berkeley.		
Reference Books			
1.	Diedrich, H., Ethereum: Blockchains, digital assets, smart contracts, decentralized autonomous organizations, 2016, 1st Edition, Wildfire publishing, Sydney.		
2.	Wattenhofer, R. P, Distributed Ledger Technology: The Science of the Blockchain (Inverted Forest Publishing), 2017, 2nd Edition, Createspace Independent Pub, Scotts Valley, California, US.		
Mode of Evaluation: CAT, written assignment, Quiz, FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE325L	INTRODUCTION TO BITCOIN	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To Identify the process of Cryptocurrency.
2. To understand the functionality of Bitcoin.
3. To explore the recent developments on Bitcoin.

Course Outcomes

After completion of this course, the student shall be able to:

1. Understand the fundamentals of Cryptography.
2. Gain knowledge about various operations associated with Cryptocurrency.
3. Develop the methods for verification and validation of Bitcoin transactions.
4. Apply the principles, practices and policies associated with Bitcoin business.

Module:1	**Fundamentals of Cryptography**	**5 hours**

Cryptographic Hash Functions - Hash Pointers and Data Structures - Digital Signatures - Public Keys as Identities - A Simple Cryptocurrency.

Module:2	**Features of Bitcoin**	**6 hours**

Bitcoin Transactions - Bitcoin Scripts - Applications of Bitcoin Scripts - Bitcoin Blocks - Bitcoin Network and Limitations.

Module:3	**Bitcoin Techniques**	**7 hours**

Techniques to Store and Use Bitcoins - Hot and Cold Storage - Splitting and Sharing Keys - Online Wallets and Exchanges - Payment Services - Transaction Fees - Bitcoin Trading.

Module:4	**Bitcoin Mining**	**8 hours**

Task of Bitcoin Miners - Mining Hardware - Energy Consumption and Ecology - Mining Pools - Mining Incentives - Merkley Tree - hardness of mining - transaction verifiability.

Module:5	**Bitcoin and Anonymity**	**5 hours**

Anonymity – Re-identification of Bitcoin - Mixing and Decentralisation of Bitcoin - Zero coin and Zero cash.

Module:6	**Mining Strategies**	**5 hours**

Essential Puzzle Requirements – Application Specific Integrated Circuit Resistant(ASIC) Puzzles - Proof of Volunteer computing - Non externalization of Puzzles - Proof of Stake Virtual Mining.

Module:7	**Bitcoin as a Platform**	**7 hours**

Bitcoin as an Append-Only Log - Bitcoin as Smart Property - Secure Multi-Party Lotteries in Bitcoin - Bitcoin as Randomness Source - Prediction Markets and Real-World Data Feeds.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book

1. Goldfeder, S., Bonneau, J., Miller, A., Felten, E., Narayanan, A. Bitcoin and Cryptocurrency Technologies, 2016, 1st edition, Princeton University Press, New Jersey.

Reference Books

1. Antonopoulos, A. M. Mastering Bitcoin: unlocking digital cryptocurrencies, 2017, 2nd edition, OReilly Media, Inc, United States.
2. Lewis, Antony, The Basics Of Bitcoins and Blockchains: An Introduction To Cryptocurrencies and The Technology That Powers Them., 2018, 1st edition, Mango Media Inc., United States.

Mode of Evaluation: CAT / Assignment / Quiz / FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE326L	BLOCKCHAIN ARCHITECTURE DESIGN	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To provide the knowledge on Blockchain architecture.
2. To understand the design of Blockchain transaction and security issues.
3. To study about various use Cases in Blockchain.

Course Outcome

After completion of this course, the student shall be able to:
1. Understand the requirements of the fundamentals of Blockchain.
2. Identify and apply the concept of Bitcoin.
3. Recognize the underlying technology of transactions, blocks and proof-of-work.
4. Gain a deep insight into Bitcoin network, Bitcoin miners and Bitcoin transactions.
5. Design and explore the applications of Blockchain.

Module:1	**Fundamentals of Blockchain**	**6 hours**

Blockchain: Importance and features – Layers of Blockchain: application layer, execution layer, semantic layer, propagation layer, consensus layer – Types of Blockchain – Blockchain in practical use today – Blockchain governance challenges – Blockchain technical challenges.

Module:2	**Blockchain for Enterprise**	**6 hours**

Blockchain Components and Concepts - Block Header and Identifiers - Linking Blocks in the Blockchain - Mining and Consensus: Aggregating transactions into Blocks - Mining the Block - Validating and Assembling of Blocks, Selecting Chains of Blocks.

Module:3	**Transactions and Bitcoin Network**	**6 hours**

Transactions: Lifecycle, Structure, Inputs and Outputs, Standard Transactions - Bitcoin Network: Network discovery for a new node, Block propagation.

Module:4	**Bitcoin Client**	**8 hours**

Consensus in Bitcoin: Proof of Work (PoW), Mining the Block, Changing the Consensus Rules - Bitcoin Core: Bitcoin core application programming interface, running a bitcoin core node, Alternative clients, libraries and toolkits - Bitcoin Addresses: Implementing Keys and Addresses in Python – Wallets.

Module:5	**Security and privacy practices**	**6 hours**

Security Architecture principles - Technical and inherent risks of the blockchain technology - Attacks on Privacy: Blockchain and non-blockchain based Attacks - Risks and Limitations of Blockchain – User security best practices: physical bitcoin storage, hardware wallets, balancing risk, diversifying risk, multi signature and governance.

Module:6	**Blockchain Architecture and Applications**	**6 hours**

Design methodology for blockchain applications: blockchain application templates, blockchain application development – Ethereum – Solidity - Deploying a sample application: Blockchain and betting – Colored coins – Counterparty.

Module:7	**Blockchain Use Cases**	**5 hours**

Blockchain in Financial Software and Systems - Supply chain and logistics monitoring - Music royalties tracking - Advertising insights - Blockchain implementation for Land Records - Digital content publishing and selling - Digital Supply chain - Medical Record Management System

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)

1. Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Beginning Blockchain, A Beginner's Guide to Building Blockchain Solutions, 2018, 1st edition, Apress, New York.
2. Joseph J. Bambara, Paul R. Allen, Blockchain: a practical guide to developing business,

	law and technology solutions, 2018, 1st edition, McGraw-Hill publication, New York.		
Reference Books			
1.	Swan Melanie, Blockchain: Blueprint for a new economy, 2015, 1st edition, O'Reilly Media, United States.		
Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE327L	SMART CONTRACTS	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the Smart Contracts in Blockchain.
2. To learn the tools and programming skills required to generate Smart Contracts.
3. To assess the efficiency of the security issues.

Course Outcomes

After completion of this course, the student shall be able to:
1. Understand the basics and objectives of Smart Contracts in a Blockchain.
2. Evaluate the various functionalities and features in an Ethereum to generate Smart Contracts.
3. Introduce the Solidity language in creation of a Smart Contracts.
4. Incorporate Smart Contracts in decentralized applications.
5. Assess the security issues and effectiveness of a Smart Contracts in real world scenarios.

Module:1	**Fundamentals of Smart Contracts**	**2 hours**
Blockchain Terminologies - Cryptocurrency and Smart Contracts - Understanding the Virtual Machine of a Blockchain - Terminology, concepts and practices in Smart Contracts.		
Module:2	**Ethereum Smart Contracts**	**5 hours**
Definition of Ethereum - Prevalence of the Ethereum blockchain in Smart Contracts development - Ethereum Virtual Machine (EVM) - Instances of working Ethereum Smart Contracts.		
Module:3	**Various Aspects in Application of Smart Contracts**	**5 hours**
Market impact and scientific innovation – Trust - Security, using Merkle Trees - Future-resistance features in Smart Contracts applications - Workflow of developing a Smart Contracts - Execution environments in writing a Smart Contracts.		
Module:4	**Solidity Language Basics**	**4 hours**
Layout of a Solidity Source File - Structure of a contracts - Control structures – Functions - Scoping and declarations.		
Module:5	**Solidity with Contracts**	**4 hours**
Creating contracts - Object-oriented high level language features - Visibility and Getters – Events - Abstract Contracts.		
Module:6	**Decentralized Applications**	**4 hours**
Decentralized Application Architecture - Connecting to the Blockchain and Smart Contracts – Building dApps – Deployment.		
Module:7	**Security Issues**	**4 hours**
Shifting from Trust-in-People to Trust-in-Code - Data permanence - Selective-Obscurity - Security counter measures.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book

1.	Gavin Zheng, Longxiang Gao, Liqun Huang, Jian Guan, Ethereum Smart Contracts Development in Solidity, 2021, 1st Edition, Springer Singapore.

Reference Books

1.	Dannen, C., Introducing Ethereum and solidity, 2017, (Vol. 318). Berkeley: Springer.
2.	Modi, Ritesh, Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and Blockchain, 2018, Packt Publishing Ltd, United Kingdom.
3.	Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder,

	Bitcoin and cryptocurrency technologies: a comprehensive introduction, 2016, Princeton University Press.			
Mode of Evaluation: CAT / written assignment / Quiz / FAT				
Recommended by Board of Studies	04-03-2022			
Approved by Academic Council	No. 65	Date	17-03-2022	

BCSE327P	SMART CONTRACTS LAB	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the Smart Contracts in Blockchain.
2. To learn the tools and programming skills required to generate Smart Contracts.
3. To assess the efficiency of the security issues.

Course Outcomes

After completion of this course, the student shall be able to:

1. Evaluate the various functionalities and features in an Ethereum to generate Smart Contracts.
2. Assess the security issues and effectiveness of a Smart Contracts in real world scenarios.

Indicative Experiments

1. Setting up Ethereum network by using Geth command line interface.
2. Identifying and setting up a testnet, like Ropsten or Kovan, so that free ethers can be used as transaction.
3. Transfer ethers from one account to another on an Ethereum testnet.
4. Constructing Solidity code for a decentralized application where the owner can create a contracts (with a tenant) which can be replicated to all nodes.
5. In a rented house setup with the owner and the tenants, the tenant can submit a deposit and the contracts's state changes on all the decentralized nodes.
6. The owner should be able to check the balance of the contracts from any one of the nodes.
7. Using Remix on the Solidity code to develop, compile and deploy the contracts.
8. Using setter and getter functions to interact with the contracts
9. Withdrawing funds from a contracts to a restricted account, preferably the owner's, with different levels of security restrictions.
10. Deploying a contracts on an external blockchain by using Ganache and/or MyEtherwalllet, Metamask.

Total Laboratory Hours	**30 hours**

Text Book

1. Gavin Zheng, Longxiang Gao, Liqun Huang, Jian Guan, Ethereum Smart Contracts Development in Solidity, 2021, 1st Edition, Springer Singapore.

Reference Books

1. Modi, Ritesh. Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and blockchain. 2018, Packt Publishing Ltd, United Kingdom.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, Bitcoin and cryptocurrency technologies: a comprehensive introduction, 2016, Princeton University Press.

Mode of assessment: Continuous assessment / FAT

Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE328L	CRYPTOCURRENCY TECHNOLOGIES	L	T	P	C
		3	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To introduce the cryptocurrency concepts and techniques used in business transactions.
2. To provide skills and knowledge about operations and management in cryptocurrency technologies applied in large scale business.
3. To develop own cryptocurrencies that meets the business and customer needs.

Course Outcome

After completion of this course, the student shall be able to:

1. Understand the evolution, principles and benefits of cryptocurrencies.
2. Assess existing technologies to choose an appropriate technology that meets business needs.
3. Implement the scripting foundations to cater the needs of generating own cryptocurrencies.
4. Decide a suitable model to capture the business needs by interpreting different crypto primitives.
5. Infer the various bitcoin related security and privacy issues and building own cryptocurrencies.

Module:1	**Fundamentals of Cryptocurrency**	**7 hours**

Cryptocurrency - Origin and Importance - Legal Status - Usage of Cryptocurrency - Blockchain Structure - Interaction between Blockchain and Cryptocurrencies - Importance and uses of Cryptocurrency - Hardware and Software requirements of Block chain.

Module:2	**Functional Aspects of Cryptocurrency**	**8 hours**

Bitcoin and other Cryptocurrencies - Distributed consensus and atomic broadcast - Alternatives to Bitcoin consensus - Alternative coins - Byzantine fault-tolerant consensus methods - Blockchain based cryptocurrency and its applications - Technologies borrowed in Blockchain.

Module:3	**Bitcoin Scripting**	**5 hours**

Bitcoin scripting language and their uses - Transactions - Signatures - Pay to script hash - Segregated Witness - Pay To Multi-signature - Storing Data - Timelocks - Hash Time-Locked Contracts - Atomic Swaps - Payment Channels.

Module:4	**Crypto Primitives for Cryptocurrency**	**5 hours**

Hash functions - Puzzle-friendly Hash - Collison resistant hash - Hash pointers and digital signatures - public key crypto - verifiable random functions - Zero-knowledge systems - Bitcoin Blockchain - Interaction with the blockchain - Elliptic curve cryptography in blockchain - SHA-256.

Module:5	**Security & Privacy Issues in Cryptocurrency**	**4 hours**

Building a Secure Bitcoin payment system - Building a Secure payment gateway - Compiling Bitcoin from source new cryptocurrency - Cloning Bitcoin - Reader coin rebranding - Securing Peer-to-Peer Auctions in Ethereum - Applications of blockchain in cyber security.

Module:6	**Building Own Cryptocurrency**	**7 hours**

Coding Own Cryptocurrency on Ethereum - Building ERC-20 Token - Integrity of information - E-Governance and other contract enforcement mechanisms - Limitations of blockchain - Myths vs. reality of blockchain technology.

Module:7	**Future Directions of Cryptocurrency**	**7 hours**

Smart Property - Efficient micro-payments - Coupling Transactions and Payment (Interdependent Transactions) - Public Randomness Source Prediction Markets - Escrow transactions - Green addresses - Auctions and Markets - Multi-party Lotteries.			
Module:8	**Contemporary Issues**		**2 hours**
	Total Lecture hours:		**45 hours**
Text Book			
1.	Daskalakis, Nikos, and Panagiotis Georgitseas. An Introduction to Cryptocurrencies: The Crypto Market Ecosystem, 2020, 1st Edition, Routledge, New York.		
Reference Books			
1.	Grabowski, Mark. Cryptocurrencies: A Primer on Digital Money, 2019, 1st Edition, Routledge, New York.		
2.	Narayanan, Arvind, et al. Bitcoin and cryptocurrency technologies: a comprehensive introduction, 2016, 1st Edition, Princeton University Press, New Jersey.		
Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE329L	BLOCKCHAIN AND DISTRIBUTED LEDGER TECHNOLOGY	L	T	P	C
		2	0	0	2
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand Blockchain and Distributed Ledger Technologies.
2. To learn the development in Blockchain functionalities.
3. To identify alternative techniques to proof of work for Blockchain protocols, proof of stake/space.

Course Outcomes

After completion of this course, the student shall be able to:

1. Comprehend the functionality of blockchain.
2. Choose a blockchain implementation based on real time scenario.
3. Examine the techniques for anonymity preservation.
4. Determine the Blockchain challenges.
5. Identify the use cases of distributed ledger technology.
6. Evaluate alternative blockchain and their applicability.

Module:1	**Blockchain and Distributed Ledger Fundamentals**	**4 hours**

Blockchain - Distributed Ledger - Cryptographic basics for cryptocurrency - signature schemes, encryption schemes and elliptic curve cryptography - CAP theorem - Categories of Blockchain: Public blockchain, Private blockchain, Permissioned Ledger, Tokenized blockchain, Tokenless blockchain, and Sidechains.

Module:2	**Blockchain Functionality**	**5 hours**

Distributed identity: Public and private keys, Digital identification and wallets - Decentralized network - Permissioned distributed Ledger - Blockchain data structure - Double spending - Network consensus - Sybil attacks - Block rewards and miners - Forks and consensus chain - Finality in Blockchain Consensus - Limitation of proof-of-work - Alternatives to Proof of Work.

Module:3	**Blockchain Implementation**	**4 hours**

Bitcoin and Merkle Root - Eventual Consistency and Bitcoin - Byzantine Fault Tolerance - Bitcoin and Secure Hashing - Bitcoin block-size - Bitcoin Mining - Blockchain Collaborative Implementations: Hyperledger, Corda - Ethereum's ERC 20 and token explosion.

Module:4	**Decentralization using Blockchain**	**4 hours**

Blockchain and full ecosystem decentralization: Smart contract, Decentralized autonomous organization (DAO), Decentralized applications - Platforms for decentralization.

Module:5	**Zero Knowledge Proofs and Protocols in Blockchain**	**4 hours**

Pseudo-anonymity vs. anonymity - Succinct non interactive argument for Knowledge (SNARK) - pairing on Elliptic curves – Zcash - Zk-SNARKS for anonymity preservation.

Module:6	**Blockchain Challenges**	**3 hours**

Blockchain Governance Challenges: Bitcoin Blocksize Debate, The Ethereum DAO Fork, Ethereum's Move to PoS and Scaling Challenges - Blockchain Technical Challenges: Denial-of-Service Attacks, Security in Smart Contracts, Scaling, Sharding.

Module:7	**Distributed Ledger Technology in Alternative Blockchain**	**4 hours**

Kadena, Ripple, Stellar, Rootstock, Drivechain, Quorum – Decentralized Network manager: Tezos, Maidsafe, BigChainDB - Decentralized Cloud Storage: Storj.

Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**30 hours**

Text Book

1. Goldfeder, S., Bonneau, J., Miller, A., Felten, E., Narayanan, A. Bitcoin and

	Cryptocurrency Technologies, 2016, 1st edition, Princeton University Press, New Jersey.			
Reference Books				
1.	Iyer, Kedar, et al. Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions., 2018, 1st edition, McGraw-Hill Education, United Kingdom.			
2.	Wattenhofer, R. Distributed Ledger Technology: The Science of the Blockchain, 2017, 1st edition, CreateSpace Independent Publishing Platform, United States.			
Mode of Evaluation: CAT / written assignment / Quiz / FAT				
Recommended by Board of Studies		04-03-2022		
Approved by Academic Council		No. 65	Date	17-03-2022

BCSE329P	**BLOCKCHAIN AND DISTRIBUTED LEDGER TECHNOLOGY LAB**	**L**	**T**	**P**	**C**
		0	**0**	**2**	**1**

Pre-requisite	**NIL**	**Syllabus version**
		1.0

Course Objectives

1. To understand Blockchain and Distributed Ledger Technologies.
2. To learn the development in Blockchain functionalities.
3. To identify alternative techniques to proof of work for Blockchain protocols, proof of stake/space.

Course Outcomes

After completion of this course, the student shall be able to:

1. Implement a blockchain for real time scenario.
2. Evaluate alternative blockchain and their applicability.

Indicative Experiments

1.	Deploy a local private blockchain over a network with Ethereum or Rust.
2.	Implement the **mining** module of Bitcoin client using Rust. The mining module, or miner, should produce blocks that solve proof-of-work puzzle.
3.	Compile and test smart contracts on a testing framework using the Ethereum Virtual Machine (EVM).
4.	Deploy a chaincode using Hyperledger Fabric on a custom network.
5.	Create a Hyperledger Fabric Blockchain service on Cloud.
6.	Deploying a ERC20 token on the Ethereum Testnet.
7.	Launch your own token on alternative blockchain such as BigchainDB

Total Laboratory Hours	**30 hours**

Text Book

1	Goldfeder, S., Bonneau, J., Miller, A., Felten, E., Narayanan, A. Bitcoin and Cryptocurrency Technologies, 2016, 1st edition, Princeton University Press, New Jersey.

Reference Books

1	Iyer, Kedar, et al. Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions., 2018, 1st edition, McGraw-Hill Education, United Kingdom.

Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE330L	PUBLIC KEY INFRASTRUCTURE AND TRUST MANAGEMENT	L	T	P	C
		3	0	0	3
Pre-requisite		**Syllabus version**			
		1.0			

Course Objectives:

1. To provide the knowledge on Public Key Cryptography techniques and Public Key infrastructure.
2. To study about the Digital Certificates and the security challenges.
3. To understand the various trust models and the trust management systems.

Course Outcome:

After completion of this course, the student shall be able to:

1. Analyze and design Public Key cryptographic algorithms.
2. Evaluate the components of PKI and design & integrate PKI services
3. Design the Digital Certificates with PKI considerations
4. Identify the access control mechanism and provide solution for the security challenges
5. Analyze and select suitable trust model and manage with operational considerations

Module:1	**Public Key Cryptography Basics**	**5 hours**

Public Key Cryptography: Secret key, Public key, public/private key pair, Services of public key cryptography - RABIN Cryptosystem - ElGamal Cryptosystem - Message Integrity and Authentication: Random Oracle model, message authentication, Cryptographic hash functions.

Module:2	**Public Key Infrastructure**	**7 hours**

Components and architecture of fully functional Public key infrastructure(PKI): Certification authority, Certificate repository, Certificate revocation, Key backup and recovery, Automatic key update, Key history management, Cross-certification, Support for non-repudiation, Time stamping, Client software, Core PKI Services, PKI-Enabled Services, PKI interoperability, deployment and assessment PKI data structures - PKI architectures: Single CA, Hierarchical PKI, Mesh PKI, Trust Lists, Bridge Certification Authority (CA), Registration Authority (RA), Simple PKI (SPKI), PKI application : Smart card integration with PKI's.

Module:3	**Digital Certificates**	**7 hours**

Introduction to Digital Certificate - Certificate Structure and Semantics - Alternative Certificate Formats - Certificate Policies - Object Identifiers - Policy Authorities - Certification Authority - Key/Certificate Life Cycle Management - Certificate Revocation - Representing certificates in terms of S-Expressions - Certificate Chain.

Module:4	**Access Control Mechanisms and Security Challenges**	**7 hours**

Access Control Mechanisms: Discretionary Access Control (DAC) – Mandatory Access Control (MAC) – Role Based Access Control (RBAC) - Issues : Revocation- Anonymity-Privacy issues - Entity Authentication - Passwords and Challenge Response - zero-knowledge and bio-metrics - Key management - security key distribution – Kerberos - Symmetric Key agreement - Public Key Distribution and Hi-jacking - Issues of revocation - Anonymity and Privacy.

Module:5	**Trust Models**	**7 hours**
Distributed Trust Architecture - Mesh Configuration - Hub-and-Spoke Configuration – Four-Corner Trust Model - Web Model - User-Centric Trust - Cross-Certification - Entity Naming - Certificate Path Processing - Path Construction - Path Validation - Trust Anchor Considerations - Multiple Key Pairs - Key Pair Uses - Relationship between Key Pairs and Certificates.		
Module:6	**Trust Management Systems**	**5 hours**
Social network based Trust Management System- Reputation based Trust Management System (DMRep, EigenRep, P2Prep) - Framework for Trust Establishment - Risks Impact on E-Commerce and E- Business: Information Risk and Technology Business Risk.		
Module:7	**Operational Considerations**	**5 hours**
Client-Side Software - Off-line Operations - Physical Security - Hardware Components - User Key Compromise - Disaster Preparation and Recovery - Relying Party Notification – Preparation – Recovery - Electronic Signature Legislation and Considerations.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Book(s)	
1.	John R. Vacca, Public Key Infrastructure: Building Trusted Applications and Web Services, 2019, 1st edition. Auerbach Publications, US.
2.	Carlisle Adams, Steve Lloyd, Understanding PKI: Concepts, Standards, and Deployment Considerations, 2011, 2nd Edition, Addison-Wesley, US.
Reference Books	
1.	Buchmann J, Karatsiolis E, Wiesmaier A, Karatsiolis E., Introduction to public key infrastructures, 2013, Berlin: Springer.

Mode of Evaluation: CAT / written assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE391J	Technical Answers to Real Problems Project	L	T	P	C
		0	0	0	3
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives:

1. To gain an understanding of real-life issues faced by society.
2. To study appropriate technologies in order to find a solution to real life issues.
3. Students will design system components intended to solve a real-life issue.

Course Outcome:

1. Identify real life issue(s) faced by society.
2. Apply appropriate technologies to suggest a solution to the identified issue(s).
3. Design the related system components/processes intended to provide a solution to the identified issue(s).

Module Content

Students are expected to perform a survey and interact with society to find out the real life issues.

Logical steps with the application of appropriate technologies should be suggested to solve the identified issues.

Subsequently the student should design the related system components or processes which is intended to provide the solution to the identified real-life issues.

General Guidelines:

1. Identification of real-life problems
2. Field visits can be arranged by the faculty concerned
3. Maximum of 3 students can form a team (within the same/different discipline)
4. Minimum of eight hours on self-managed team activity
5. Appropriate scientific methodologies to be utilized to solve the identified issue
6. Solution should be in the form of fabrication/coding/modelling/product design/process design/relevant scientific methodology(ies)
7. Consolidated report to be submitted for assessment
8. Participation, involvement and contribution in group discussions during the contact hours will be used as the modalities for the continuous assessment of the theory component
9. Project outcome to be evaluated in terms of technical, economical, social, environmental, political and demographic feasibility
10. Contribution of each group member to be assessed

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – Report to be submitted, presentation and project reviews

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE392J	Design Project	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. Students will be able to upgrade a prototype to a design prototype.
2. Describe and demonstrate the techniques and skills necessary for the project.
3. Acquire knowledge and better understanding of design systems.

Course Outcome:

1. Develop new skills and demonstrate the ability to upgrade a prototype to a design prototype or working model.
2. Utilize the techniques, skills, and modern tools necessary for the project.
3. Synthesize knowledge and use insight and creativity to better understand and improve design systems.

Module Content

Students are expected to develop new skills and demonstrate the ability to develop prototypes to design prototype or working models related to an engineering product or a process.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – Report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE393J	Laboratory Project	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. The student will be able to conduct experiments on the concepts already learnt.
2. Analyse experimental data.
3. Present the results with appropriate interpretation.

Course Outcome:

1. Design and conduct experiments in order to gain hands-on experience on the concepts already studied.
2. Analyse and interpret experimental data.
3. Write clear and concise technical reports and research articles

Module Content

Students are expected to perform experiments and gain hands-on experience on the theory courses they have already studied or registered in the ongoing semester. The theory course registered is not expected to have laboratory component and the student is expected to register with the same faculty who handled the theory course. This is mostly applicable to the elective courses. The nature of the laboratory experiments is depended on the course.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – Report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE394J	Product Development Project	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. Students will be able to translate a prototype to a useful product.
2. Apply relevant codes and standards during product development.
3. The student will be able to present his results by means of clear technical reports.

Course Outcome:

1. Demonstrate the ability to translate the developed prototype/working model to a viable product useful to society/industry.
2. Apply the appropriate codes/regulations/standards during product development.
3. Write clear and concise technical reports and research articles

Module Content

Students are expected to translate the developed prototypes / working models into a product which has application to society or industry.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – Report to be submitted, presentation and project reviews

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE396J	Reading Course	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. The student will be able to analyse and interpret published literature for information pertaining to niche areas.
2. Scrutinize technical literature and arrive at conclusions.
3. Use insight and creativity for a better understanding of the domain of interest.

Course Outcome:

1. Retrieve, analyse, and interpret published literature/books providing information related to niche areas/focused domains.
2. Examine technical literature, resolve ambiguity, and develop conclusions.
3. Synthesize knowledge and use insight and creativity to better understand the domain of interest.

Module Content

This is oriented towards reading published literature or books related to niche areas or focussed domains under the guidance of a faculty.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – Report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE397J	Special Project	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. Students will be able to identify and solve problems in a time-bound manner.
2. Describe major approaches and findings in the area of interest.
3. Present the results in a clear and concise manner.

Course Outcome:

1. To identify, formulate, and solve problems using appropriate information and approaches in a time-bound manner.
2. To demonstrate an understanding of major approaches, concepts, and current research findings in the area of interest.
3. Write clear and concise research articles for publication in conference proceedings/peer-reviewed journals.

Module Content

This is an open-ended course in which the student is expected to work on a time bound research project under the supervision of a faculty. The result may be a tangible output in terms of publication of research articles in a conference proceeding or in a peer-reviewed Scopus indexed journal.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – project report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE398J	Simulation Project	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. Students will be able to simulate a real system.
2. Identify the variables which affect the system.
3. Describe the performance of a real system.

Course Outcome:

1. Demonstrate the ability to simulate and critically analyse the working of a real system.
2. Identify and study the different variables which affect the system elaborately.
3. Evaluate the impact and performance of the real system.

Module Content

The student is expected to simulate and critically analyse the working of a real system. Role of different variables which affect the system has to be studied extensively such that the impact of each step in the process is understood, thereby the performance of each step of the engineering process is evaluated.

Mode of Evaluation: Evaluation involves periodic reviews by the faculty with whom the student has registered. Assessment on the project – Mark weightage of 20:30:50 – project report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BEEE303L	Control Systems	L	T	P	C
		3	0	0	3
Pre-requisites	**BEEE101L, BEEE101P, BMAT102L**	**Syllabus version**			
		1.0			

Course Objectives

1. Introduce the fundamentals of physical systems modelling and control of linear time invariant systems.
2. Teach the practical control system design with realistic system specifications.
3. Impart knowledge of state variable models and state feedback design.

Course Outcome

On the completion of this course, the student will be able to:
1. Formulate mathematical models of the physical systems.
2. Analyze the system performance in time and frequency domains.
3. Determine the stability of linear time invariant system in time and frequency domains.
4. Design compensators and controllers to meet the performance specifications.
5. Perform state space analysis and design state feedback control.

Module:1	**Systems and their Representations**	**6 hours**
Basic elements in control systems: open loop and closed loop, transfer functions of mechanical, electrical and electro-mechanical systems, electrical analogous systems; Block diagram reduction, signal flow graphs.		
Module:2	**Time Response Analysis**	**6 hours**
Standard test signals, time response of first and second order systems, time domain specifications; Steady state error, static error constants and system type.		
Module:3	**Stability Analysis and Root Locus**	**6 hours**
Stability: concept and definition, characteristic equation, location of poles, Routh Hurwitz criterion; Root locus technique: construction, properties and applications.		
Module:4	**Frequency Response Analysis**	**6 hours**
Frequency domain specifications; Bode plot, Polar plot; Correlation between frequency domain and time domain specifications.		
Module:5	**Stability in Frequency Domain**	**5 hours**
Relative stability: gain margin, phase margin; stability analysis using frequency response methods; Nyquist stability criterion.		
Module:6	**Compensators and Controllers**	**7 hours**
Realization of basic compensators, cascade compensation in time domain and frequency domain, feedback compensation, design of lag, lead, lag-lead series compensators using Bode plot; P, PI and PID controllers in frequency domain.		
Module:7	**State Space Analysis**	**7 hours**
Concepts of state variable and state model, solution of state equation, state space to transfer function conversion, state space decomposition methods, controllability, observability, pole placement control, observer design.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

Text Books

1. Norman S. Nise, Control System Engineering, 2019, 8th Edition, John Wiley & Sons
2. Farid Galnaraghi, Benjamin C. Kuo, Automatic Control System, 2017, 9th Edition, McGraw-Hill Education

Reference Books

1. K. Ogata, Modern Control Engineering, 2016, 5th Edition, Pearson
2. R.C. Dorf & R.H. Bishop, Modern Control Systems, 2017, 13th Edition, Pearson

	Education			
3.	M. Gopal, Control Systems- Principles and Design, 2016, 4th Edition, Tata McGraw Hill			
4.	J. Nagrath and M. Gopal, Control System Engineering, 2018, 6th Edition, New Age International Publishers			
Mode of Evaluation: CAT, Assignment, Quiz, FAT				
Recommended by Board of Studies		19-02-2022		
Approved by Academic Council		No. 65	Date	17-03-2022

BEEE303P	Control Systems Lab	L	T	P	C
		0	0	2	1
Pre-requisites	**BEEE101L, BEEE101P, BMAT102L**	**Syllabus version**			
		1.0			

Course Objectives

1. Develop transfer function and state space models of physical systems.
2. Design and implement a PID controller/State feedback controller/ Lag/Lead/Lag-lead compensators.

Course Outcomes

On the completion of this course, the student will be able to:
1. Design feedback control for meeting system specifications.
2. Analyze the stability and response of linear time invariant systems.
3. Perform the time and frequency domain analyses of first and second order systems.

Indicative Experiments

1.	Simulation study of block diagram reduction technique
2.	Determination of time domain specifications
3.	Study of first and second order electrical networks
4.	Stability analysis of linear systems
5.	PID controller design using Bode plot
6.	PID controller design using root locus
7.	Compensator design in frequency and time domains
8.	Analysis of controllability and observability properties of a system
9.	Lag compensator design for linear servo motor for speed control application
10.	Pole placement controller design for inverted pendulum
11.	PD controller design for position control of servo plant
12.	Cascade control design for ball and beam system
13.	PID controller design for magnetic levitation system
14.	Determination of transfer function of separately excited DC generator
15.	Identification of transfer function of field-controlled separately excited DC Motor
16.	Controller realization from MATLAB / SIMULINK using Embedded Coder
	Total Laboratory Hours | 30 hours

Mode of assessment: Continuous assessment, FAT

Text Book

1. Norman S. Nise, Control System Engineering, 2019, 8th Edition, John Wiley & Sons

Recommended by Board of Studies	19-02-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

Project and Internship

BCSE399J	Summer Industrial Internship	L	T	P	C
		0	0	0	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

1. The course is designed so as to expose the students to industry environment and to take up on-site assignment as trainees or interns.

Course Outcome:

1. Demonstrate professional and ethical responsibility.
2. Understand the impact of engineering solutions in a global, economic, environmental and societal context.
3. Develop the ability to engage in research and to involve in life-long learning.
4. Comprehend contemporary issues.

Module Content

Four weeks of work at industry site.
Supervised by an expert at the industry.

Mode of Evaluation: Internship Report, Presentation and Project Review

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE497J	Project - I	L	T	P	C
		0	0	0	3
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

To provide sufficient hands-on learning experience related to the design, development and analysis of suitable product / process so as to enhance the technical skill sets in the chosen field.

Course Outcome:

1. Demonstrate professional and ethical responsibility.
2. Evaluate evidence to determine and implement best practice.
3. Mentor and support peers to achieve excellence in practice of the discipline.
4. Work in multi-disciplinary teams and provide solutions to problems that arise in multi-disciplinary work.

Module Content

Project may be a theoretical analysis, modeling & simulation, experimentation & analysis, prototype design, fabrication of new equipment, correlation and analysis of data, software development, applied research and any other related activities.

Can be individual work or a group project, with a maximum of 3 students.

In case of group projects, the individual project report of each student should specify the individual's contribution to the group project.

Carried out inside or outside the university, in any relevant industry or research institution.

Publications in the peer reviewed journals / International Conferences will be an added advantage.

Mode of Evaluation: Assessment on the project - project report to be submitted, presentation and project reviews

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022

BCSE498J	Project – II / Internship	L	T	P	C
		0	0	0	5
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives:

To provide sufficient hands-on learning experience related to the design, development and analysis of suitable product / process so as to enhance the technical skill sets in the chosen field.

Course Outcome:

1. Formulate specific problem statements for well-defined real life problems with reasonable assumptions and constraints.
2. Perform literature search and / or patent search in the area of interest.
3. Conduct experiments / Design and Analysis / solution iterations and document the results.
4. Perform error analysis / benchmarking / costing.
5. Synthesize the results and arrive at scientific conclusions / products / solution.
6. Document the results in the form of technical report / presentation.

Module Content

1. Project may be a theoretical analysis, modeling & simulation, experimentation & analysis, prototype design, fabrication of new equipment, correlation and analysis of data, software development, applied research and any other related activities.
2. Project can be for one or two semesters based on the completion of required number of credits as per the academic regulations.
3. Can be individual work or a group project, with a maximum of 3 students.
4. In case of group projects, the individual project report of each student should specify the individual's contribution to the group project.
5. Carried out inside or outside the university, in any relevant industry or research institution.
6. Publications in the peer reviewed journals / International Conferences will be an added advantage.

Mode of Evaluation: : Assessment on the project - project report to be submitted, presentation and project reviews.

Recommended by Board of Studies	09-03-2022		
Approved by Academic Council	No. 65	Date	17-03-2022