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 with Specialization in Data Science

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-CSE (Spl. in Data Science)

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.

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

B.Tech-CSE (Spl. in Data Science)

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-CSE (Spl. in Data Science)

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 – CSE with specialization in Data Science

Curriculum for 2021-2022 Batch

CREDIT INFO		
S.no	**Catagory**	**Credits**
1	Foundation Core	55
2	Discipline-linked Engineering Sciences	12
3	Discipline Core	44
4	Specialization Elective	21
5	Projects and Internship	9
6	Open Elective	9
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**	**Credits**
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
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	Credits
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	Credits
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

Specialization Elective

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

Specialization 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	BCSE331L	Exploratory Data Analysis	Theory Only	1.0	2	0	0	0	2.0
9	BCSE331P	Exploratory Data Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
10	BCSE332L	Deep Learning	Theory Only	1.0	3	0	0	0	3.0
11	BCSE332P	Deep Learning Lab	Lab Only	1.0	0	0	2	0	1.0
12	BCSE333L	Statistical Inference	Theory Only	1.0	2	0	0	0	2.0
13	BCSE333P	Statistical Inference Lab	Lab Only	1.0	0	0	2	0	1.0
14	BCSE334L	Predictive Analytics	Theory Only	1.0	3	0	0	0	3.0
15	BCSE335L	Healthcare Data Analytics	Theory Only	1.0	3	0	0	0	3.0
16	BCSE336L	Financial Data Analytics	Theory Only	1.0	2	0	0	0	2.0
17	BCSE336P	Financial Data Analytics 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	Credits
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	Credits
1	BCSE351E	Foundation of Data Analytics	Embedded Theory and Lab	1.0	1	0	2	0	2.0
2	BCSE352E	Essentials of Data Analytics	Embedded Theory and Lab	1.0	1	0	2	0	2.0
3	BSTS301P	Advanced Competitive Coding - I	Soft Skill	1.0	0	0	3	0	1.5
4	BSTS302P	Advanced Competitive Coding - II	Soft Skill	1.0	0	0	3	0	1.5
5	CFOC102M	Introduction to Cognitive Psychology	Online Course	1.0	0	0	0	0	3.0
6	CFOC103M	Introduction to Political Theory	Online Course	1.0	0	0	0	0	3.0
7	CFOC104M	Six Sigma	Online Course	1.0	0	0	0	0	3.0
8	CFOC105M	Emotional Intelligence	Online Course	1.0	0	0	0	0	2.0
9	CFOC109M	Design Thinking - A Primer	Online Course	1.0	0	0	0	0	1.0
10	CFOC112M	Sociology of Science	Online Course	1.0	0	0	0	0	1.0
11	CFOC118M	Practical Machine Learning with Tensorflow	Online Course	1.0	0	0	0	0	2.0

Open Elective									
12	CFOC133M	E-Business	Online Course	1.0	0	0	0	0	3.0
13	CFOC152M	Pattern Recognition and Application	Online Course	1.0	0	0	0	0	3.0
14	CFOC165M	Software testing	Online Course	1.0	0	0	0	0	3.0
15	CFOC188M	Ethical Hacking	Online Course	1.0	0	0	0	0	3.0
16	CFOC190M	Positive Psychology	Online Course	1.0	0	0	0	0	2.0
17	CFOC191M	Forests and their Management	Online Course	1.0	0	0	0	0	3.0
18	CFOC193M	Bioengineering: An Interface with Biology and Medicine	Online Course	1.0	0	0	0	0	2.0
19	CFOC197M	Bio-Informatics: Algorithms and Applications	Online Course	1.0	0	0	0	0	3.0
20	CFOC203M	Natural Hazards	Online Course	1.0	0	0	0	0	2.0
21	CFOC207M	Electronic Waste Management - Issues And Challenges	Online Course	1.0	0	0	0	0	1.0
22	CFOC227M	GPU Architectures and Programming	Online Course	1.0	0	0	0	0	3.0
23	CFOC232M	Consumer Behaviour	Online Course	1.0	0	0	0	0	2.0
24	CFOC235M	Rocket Propulsion	Online Course	1.0	0	0	0	0	3.0
25	CFOC236M	Aircraft Maintenance	Online Course	1.0	0	0	0	0	1.0
26	CFOC253M	Plastic Waste Management	Online Course	1.0	0	0	0	0	2.0
27	CFOC257M	Earthquake Geology: A tool for Seismic Hazard Assessment	Online Course	1.0	0	0	0	0	3.0
28	CFOC258M	Introduction to Geographic Information Systems	Online Course	1.0	0	0	0	0	1.0
29	CFOC282M	Waste to Energy Conversion	Online Course	1.0	0	0	0	0	2.0
30	CFOC329M	Design, Technology and Innovation	Online Course	1.0	0	0	0	0	2.0
31	CFOC332M	Fundamentals of Automotive Systems	Online Course	1.0	0	0	0	0	3.0
32	CFOC356M	Analog Circuits	Online Course	1.0	0	0	0	0	3.0
33	CFOC365M	Evolution of Air Interface towards 5G	Online Course	1.0	0	0	0	0	2.0
34	CFOC384M	Entrepreneurship Essentials	Online Course	1.0	0	0	0	0	3.0
35	CFOC388M	Energy Resources, Economics and Environment	Online Course	1.0	0	0	0	0	3.0
36	CFOC391M	Effective Writing	Online Course	1.0	0	0	0	0	1.0
37	CFOC395M	Speaking Effectively	Online Course	1.0	0	0	0	0	2.0
38	CFOC397M	Intellectual Property	Online Course	1.0	0	0	0	0	3.0
39	CFOC400M	Language and Mind	Online Course	1.0	0	0	0	0	2.0
40	CFOC401M	The Nineteenth - Century English Novel	Online Course	1.0	0	0	0	0	3.0
41	CFOC402M	Introduction to World Literature	Online Course	1.0	0	0	0	0	3.0
42	CFOC405M	Economic Growth & Development	Online Course	1.0	0	0	0	0	2.0
43	CFOC406M	Human Behaviour	Online Course	1.0	0	0	0	0	2.0
44	CFOC407M	Introduction to Modern Indian Political Thought	Online Course	1.0	0	0	0	0	3.0
45	CFOC408M	English Literature of the Romantic Period, 1798 - 1832	Online Course	1.0	0	0	0	0	2.0
46	CFOC416M	Feminism : Concepts and Theories	Online Course	1.0	0	0	0	0	3.0
47	CFOC419M	Basic Real Analysis	Online Course	1.0	0	0	0	0	3.0
48	CFOC442M	Robotics and Control : Theory and Practice	Online Course	1.0	0	0	0	0	2.0
49	CFOC475M	IC Engines and Gas Turbines	Online Course	1.0	0	0	0	0	3.0
50	CFOC488M	Business Analytics For Management Decision	Online Course	1.0	0	0	0	0	3.0
51	CFOC490M	Sales and Distribution Management	Online Course	1.0	0	0	0	0	2.0

Open Elective									
52	CFOC493M	Management of Inventory Systems	Online Course	1.0	0	0	0	0	3.0
53	CFOC494M	Quality Design And Control	Online Course	1.0	0	0	0	0	3.0
54	CFOC495M	Foundation Course in Managerial Economics	Online Course	1.0	0	0	0	0	2.0
55	CFOC496M	Engineering Econometrics	Online Course	1.0	0	0	0	0	3.0
56	CFOC497M	Financial Statement Analysis and Reporting	Online Course	1.0	0	0	0	0	3.0
57	CFOC498M	Business Statistics	Online Course	1.0	0	0	0	0	3.0
58	CFOC499M	Global Marketing Management	Online Course	1.0	0	0	0	0	2.0
59	CFOC500M	Marketing Research and Analysis - II	Online Course	1.0	0	0	0	0	3.0
60	CFOC503M	Marketing Analytics	Online Course	1.0	0	0	0	0	3.0
61	CFOC505M	Management of Commercial Banking	Online Course	1.0	0	0	0	0	3.0
62	CFOC508M	Entrepreneurship	Online Course	1.0	0	0	0	0	3.0
63	CFOC543M	International Business	Online Course	1.0	0	0	0	0	3.0
64	CFOC550M	Numerical Analysis	Online Course	1.0	0	0	0	0	4.0
65	CFOC570M	Public Speaking	Online Course	1.0	0	0	0	0	3.0
66	CFOC575M	Wildlife Ecology	Online Course	1.0	0	0	0	0	3.0
67	CFOC578M	Wastewater Treatment And Recycling	Online Course	1.0	0	0	0	0	3.0
68	CFOC591M	Principles Of Management	Online Course	1.0	0	0	0	0	3.0
69	CFOC593M	Corporate Finance	Online Course	2.0	0	0	0	0	2.0
70	CFOC594M	Customer Relationship Management	Online Course	1.0	0	0	0	0	2.0
71	CFOC595M	Urbanization and Environment	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**	**Credits**
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**	**Credits**
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

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

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

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