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 Information Security

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 Information Security)

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.

B. Tech. - CSE (Spl. in Information Security)

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 recognize 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 Information Security)

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.

VIT®

Vellore Institute of Technology

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

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
B. Tech. – CSE with specialization in Information Security

Curriculum for 2021-2022 Batch

CREDIT INFO		
S.no	**Catagory**	**Credit**
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	Credit
1	BCHY101L	Engineering Chemistry	Theory Only	1.0	3	0	0	0	3.0
2	BCHY101P	Engineering Chemistry Lab	Lab Only	1.0	0	0	2	0	1.0
3	BCSE101E	Computer Programming: Python	Embedded Theory and Lab	1.0	1	0	4	0	3.0
4	BCSE102L	Structured and Object-Oriented Programming	Theory Only	1.0	2	0	0	0	2.0
5	BCSE102P	Structured and Object-Oriented Programming Lab	Lab Only	1.0	0	0	4	0	2.0
6	BCSE103E	Computer Programming: Java	Embedded Theory and Lab	1.0	1	0	4	0	3.0
7	BECE101L	Basic Electronics	Theory Only	1.0	2	0	0	0	2.0
8	BECE101P	Basic Electronics Lab	Lab Only	1.0	0	0	2	0	1.0
9	BEEE101L	Basic Electrical Engineering	Theory Only	1.0	2	0	0	0	2.0
10	BEEE101P	Basic Electrical Engineering Lab	Lab Only	1.0	0	0	2	0	1.0
11	BENG101L	Technical English Communication	Theory Only	1.0	2	0	0	0	2.0
12	BENG101P	Technical English Communication Lab	Lab Only	1.0	0	0	2	0	1.0
13	BENG102P	Technical Report Writing	Lab Only	1.0	0	0	2	0	1.0
14	BFLE200L	B.Tech. Foreign Language - 2021	Basket	1.0	0	0	0	0	2.0
15	BHSM200L	B.Tech. HSM Elective - 2021	Basket	1.0	0	0	0	0	3.0
16	BMAT101L	Calculus	Theory Only	1.0	3	0	0	0	3.0

Foundation Core									
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	Credit
1	BECE102L	Digital Systems Design	Theory Only	1.0	3	0	0	0	3.0
2	BECE102P	Digital Systems Design Lab	Lab Only	1.0	0	0	2	0	1.0
3	BECE204L	Microprocessors and Microcontrollers	Theory Only	1.0	3	0	0	0	3.0
4	BECE204P	Microprocessors and Microcontrollers Lab	Lab Only	1.0	0	0	2	0	1.0
5	BMAT205L	Discrete Mathematics and Graph Theory	Theory Only	1.0	3	1	0	0	4.0

Discipline Core

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

Specialization Elective									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	BCSE317L	Information Security	Theory Only	1.0	3	0	0	0	3.0
2	BCSE318L	Data Privacy	Theory Only	1.0	3	0	0	0	3.0
3	BCSE319L	Penetration Testing and Vulnerability Analysis	Theory Only	1.0	2	0	0	0	2.0
4	BCSE319P	Penetration Testing and Vulnerability Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
5	BCSE320L	Web Application Security	Theory Only	1.0	3	0	0	0	3.0
6	BCSE321L	Malware Analysis	Theory Only	1.0	2	0	0	0	2.0
7	BCSE321P	Malware Analysis Lab	Lab Only	1.0	0	0	2	0	1.0
8	BCSE322L	Digital Forensics	Theory Only	1.0	2	0	0	0	2.0
9	BCSE322P	Digital Forensics Lab	Lab Only	1.0	0	0	2	0	1.0
10	BCSE323L	Digital Watermarking and Steganography	Theory Only	1.0	3	0	0	0	3.0

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

Open Elective									
sl.no	Course Code	Course Title	Course Type	Version	L	T	P	J	Credit
1	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	BCSE391J	Technical Answers to Real Problems Project	Project	1.0	0	0	0	0	3.0
4	BCSE392J	Design Project	Project	1.0	0	0	0	0	3.0
5	BCSE393J	Laboratory Project	Project	1.0	0	0	0	0	3.0
6	BCSE394J	Product Development Project	Project	1.0	0	0	0	0	3.0
7	BCSE396J	Reading Course	Project	1.0	0	0	0	0	3.0
8	BCSE397J	Special Project	Project	1.0	0	0	0	0	3.0
9	BCSE398J	Simulation Project	Project	1.0	0	0	0	0	3.0
10	BSTS301P	Advanced Competitive Coding - I	Soft Skill	1.0	0	0	3	0	1.5
11	BSTS302P	Advanced Competitive Coding - II	Soft Skill	1.0	0	0	3	0	1.5
12	CFOC102M	Introduction to Cognitive Psychology	Online Course	1.0	0	0	0	0	3.0
13	CFOC103M	Introduction to Political Theory	Online Course	1.0	0	0	0	0	3.0
14	CFOC104M	Six Sigma	Online Course	1.0	0	0	0	0	3.0
15	CFOC105M	Emotional Intelligence	Online Course	1.0	0	0	0	0	2.0
16	CFOC109M	Design Thinking - A Primer	Online Course	1.0	0	0	0	0	1.0
17	CFOC118M	Practical Machine Learning with Tensorflow	Online Course	1.0	0	0	0	0	2.0
18	CFOC122M	Educational Leadership	Online Course	1.0	0	0	0	0	2.0
19	CFOC133M	E-Business	Online Course	1.0	0	0	0	0	3.0
20	CFOC152M	Pattern Recognition and Application	Online Course	1.0	0	0	0	0	3.0
21	CFOC165M	Software testing	Online Course	1.0	0	0	0	0	3.0
22	CFOC188M	Ethical Hacking	Online Course	1.0	0	0	0	0	3.0
23	CFOC190M	Positive Psychology	Online Course	1.0	0	0	0	0	2.0
24	CFOC191M	Forests and their Management	Online Course	1.0	0	0	0	0	3.0

Open Elective									
25	CFOC193M	Bioengineering: An Interface with Biology and Medicine	Online Course	1.0	0	0	0	0	2.0
26	CFOC197M	Bio-Informatics: Algorithms and Applications	Online Course	1.0	0	0	0	0	3.0
27	CFOC203M	Natural Hazards	Online Course	1.0	0	0	0	0	2.0
28	CFOC207M	Electronic Waste Management - Issues And Challenges	Online Course	1.0	0	0	0	0	1.0
29	CFOC227M	GPU Architectures and Programming	Online Course	1.0	0	0	0	0	3.0
30	CFOC232M	Consumer Behaviour	Online Course	1.0	0	0	0	0	2.0
31	CFOC235M	Rocket Propulsion	Online Course	1.0	0	0	0	0	3.0
32	CFOC236M	Aircraft Maintenance	Online Course	1.0	0	0	0	0	1.0
33	CFOC253M	Plastic Waste Management	Online Course	1.0	0	0	0	0	2.0
34	CFOC257M	Earthquake Geology: A tool for Seismic Hazard Assessment	Online Course	1.0	0	0	0	0	3.0
35	CFOC258M	Introduction to Geographic Information Systems	Online Course	1.0	0	0	0	0	1.0
36	CFOC282M	Waste to Energy Conversion	Online Course	1.0	0	0	0	0	2.0
37	CFOC329M	Design, Technology and Innovation	Online Course	1.0	0	0	0	0	2.0
38	CFOC332M	Fundamentals of Automotive Systems	Online Course	1.0	0	0	0	0	3.0
39	CFOC356M	Analog Circuits	Online Course	1.0	0	0	0	0	3.0
40	CFOC365M	Evolution of Air Interface towards 5G	Online Course	1.0	0	0	0	0	2.0
41	CFOC384M	Entrepreneurship Essentials	Online Course	1.0	0	0	0	0	3.0
42	CFOC388M	Energy Resources, Economics and Environment	Online Course	1.0	0	0	0	0	3.0
43	CFOC391M	Effective Writing	Online Course	1.0	0	0	0	0	1.0
44	CFOC395M	Speaking Effectively	Online Course	1.0	0	0	0	0	2.0
45	CFOC397M	Intellectual Property	Online Course	1.0	0	0	0	0	3.0
46	CFOC400M	Language and Mind	Online Course	1.0	0	0	0	0	2.0
47	CFOC401M	The Nineteenth - Century English Novel	Online Course	1.0	0	0	0	0	3.0
48	CFOC402M	Introduction to World Literature	Online Course	1.0	0	0	0	0	3.0
49	CFOC405M	Economic Growth & Development	Online Course	1.0	0	0	0	0	2.0
50	CFOC406M	Human Behaviour	Online Course	1.0	0	0	0	0	2.0
51	CFOC407M	Introduction to Modern Indian Political Thought	Online Course	1.0	0	0	0	0	3.0
52	CFOC408M	English Literature of the Romantic Period, 1798 - 1832	Online Course	1.0	0	0	0	0	2.0
53	CFOC416M	Feminism : Concepts and Theories	Online Course	1.0	0	0	0	0	3.0
54	CFOC419M	Basic Real Analysis	Online Course	1.0	0	0	0	0	3.0
55	CFOC442M	Robotics and Control : Theory and Practice	Online Course	1.0	0	0	0	0	2.0
56	CFOC475M	IC Engines and Gas Turbines	Online Course	1.0	0	0	0	0	3.0
57	CFOC488M	Business Analytics For Management Decision	Online Course	1.0	0	0	0	0	3.0
58	CFOC490M	Sales and Distribution Management	Online Course	1.0	0	0	0	0	2.0
59	CFOC493M	Management of Inventory Systems	Online Course	1.0	0	0	0	0	3.0
60	CFOC494M	Quality Design And Control	Online Course	1.0	0	0	0	0	3.0
61	CFOC495M	Foundation Course in Managerial Economics	Online Course	1.0	0	0	0	0	2.0
62	CFOC496M	Engineering Econometrics	Online Course	1.0	0	0	0	0	3.0
63	CFOC497M	Financial Statement Analysis and Reporting	Online Course	1.0	0	0	0	0	3.0
64	CFOC498M	Business Statistics	Online Course	1.0	0	0	0	0	3.0
65	CFOC499M	Global Marketing Management	Online Course	1.0	0	0	0	0	2.0
66	CFOC500M	Marketing Research and Analysis - II	Online Course	1.0	0	0	0	0	3.0
67	CFOC503M	Marketing Analytics	Online Course	1.0	0	0	0	0	3.0
68	CFOC505M	Management of Commercial Banking	Online Course	1.0	0	0	0	0	3.0
69	CFOC508M	Entrepreneurship	Online Course	1.0	0	0	0	0	3.0
70	CFOC543M	International Business	Online Course	1.0	0	0	0	0	3.0
71	CFOC550M	Numerical Analysis	Online Course	1.0	0	0	0	0	4.0
72	CFOC570M	Public Speaking	Online Course	1.0	0	0	0	0	3.0
73	CFOC574M	Micro irrigation Engineering	Online Course	1.0	0	0	0	0	3.0
74	CFOC575M	Wildlife Ecology	Online Course	1.0	0	0	0	0	3.0
75	CFOC576M	Integrated Waste Management For A Smart City	Online Course	1.0	0	0	0	0	3.0
76	CFOC578M	Wastewater Treatment And Recycling	Online Course	1.0	0	0	0	0	3.0

Open Elective

77	CFOC581M	Algorithmic Game Theory	Online Course	1.0	0	0	0	0	3.0
78	CFOC586M	Spoken Sanskrit: Basic and Intermediate Levels	Online Course	1.0	0	0	0	0	3.0
79	CFOC589M	Engineering Drawing And Computer Graphics	Online Course	1.0	0	0	0	0	3.0
80	CFOC591M	Principles Of Management	Online Course	1.0	0	0	0	0	3.0
81	CFOC592M	Stress Management	Online Course	1.0	0	0	0	0	1.0
82	CFOC593M	Corporate Finance	Online Course	1.0	0	0	0	0	3.0
83	CFOC594M	Customer Relationship Management	Online Course	1.0	0	0	0	0	2.0

Bridge Course

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

Non-graded Core Requirement

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

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 kth 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++, 4th 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, 1^{st} 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, 1^{st} 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

<table>
<tr><td colspan="2"><b>BCSE317L</b></td><td colspan="2"><b>INFORMATION SECURITY</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>C</b></td></tr>
<tr><td colspan="2"></td><td colspan="2"></td><td><b>3</b></td><td><b>0</b></td><td><b>0</b></td><td><b>3</b></td></tr>
<tr><td colspan="2"><b>Pre-requisite</b></td><td colspan="2"></td><td colspan="4"><b>Syllabus version</b></td></tr>
<tr><td colspan="2"></td><td colspan="2"></td><td colspan="4">1.0</td></tr>
<tr><td colspan="8"><b>Course Objectives</b></td></tr>
<tr><td colspan="8">1. To learn various threats and attacks in a network.
2. To understand and explore fundamental techniques in developing secure applications.
3. To learn various methodologies for securing information systems ranging from operating systems to database management systems and to applications.</td></tr>
<tr><td colspan="8"><b>Course Outcomes</b></td></tr>
<tr><td colspan="8">After completion of this course, the student shall be able to:

1. Apply fundamental knowledge on key security concepts, access control and authentication.
2. Comprehend the use of security techniques for securing the information.
3. Apply various data privacy policies in different areas of web based security systems.
4. Differentiate the needs and application of security in Operating System and Firewalls.
5. Analyze various method of securing databases.</td></tr>
<tr><td colspan="8"></td></tr>
<tr><td><b>Module:1</b></td><td colspan="2"><b>Information Security Concepts</b></td><td colspan="5"><b>4 hours</b></td></tr>
<tr><td colspan="8">Information Security - Computer Security - Threats - Harm - Vulnerabilities - Program Security - Malicious code - Malwares: Viruses, Trojan Horses and Worms - Counter measures.</td></tr>
<tr><td><b>Module:2</b></td><td colspan="2"><b>Authentication and Access Control</b></td><td colspan="5"><b>6 hours</b></td></tr>
<tr><td colspan="8">Authentication - Key management schemes - Hierarchical Key Management Techniques - Security Standards - User Authentication Protocols - Implementing Access Controls - Access Control Models - Role Based Access Control - Attribute Based Access Control - Attribute based Encryption in Information Storage - Physical Access Controls.</td></tr>
<tr><td><b>Module:3</b></td><td colspan="2"><b>Operating Systems Security</b></td><td colspan="5"><b>7 hours</b></td></tr>
<tr><td colspan="8">Security in Operating System - Security in the design of OS: Simplified Design, Layered Design, Kernelized design, Reference Monitor, Trusted Systems, Trusted Systems Functions - Trusted Operating System Design - Rootkit.</td></tr>
<tr><td><b>Module:4</b></td><td colspan="2"><b>Security Countermeasures</b></td><td colspan="5"><b>7 hours</b></td></tr>
<tr><td colspan="8">Design of Firewalls - Types - Personal Firewalls - Configurations - Network Address Translation - Data Loss Prevention - Intrusion Detection and Prevention Systems: Types of IDSs, Intrusion Prevention system, Intrusion Response, Goals of IDSs, Strength and Limitations.</td></tr>
<tr><td><b>Module:5</b></td><td colspan="2"><b>Database Security</b></td><td colspan="5"><b>6 hours</b></td></tr>
<tr><td colspan="8">Database Security - Database Security Requirements - Reliability and Integrity - Sensitive Data - Types of Disclosures - Preventing Disclosures - Inference - Multilevel Databases - Multilevel Security - Database Attacks - SQL Injection Attacks.</td></tr>
<tr><td><b>Module:6</b></td><td colspan="2"><b>Web Security</b></td><td colspan="5"><b>6 hours</b></td></tr>
<tr><td colspan="8">Browser Attacks: Types, Failed Identification and Authentication - Misleading and Malicious Web Contents - Protection against Malicious Web Pages - Website Data: Code within Data, Cross Site Scripting Attacks - Prevention of Data Attacks - Fake e-mails - Spam Detection - Phishing Attacks - Phishing URL Detection and Prevention.</td></tr>
<tr><td><b>Module:7</b></td><td colspan="2"><b>Privacy Issues</b></td><td colspan="5"><b>7 hours</b></td></tr>
<tr><td colspan="8">Privacy Concepts: Aspects of Information Privacy, Computer-Related Privacy Problems - Threats to Personal Data Privacy - People-Based Privacy Concerns - Privacy Principles and Policies - Individual Actions to Protect Privacy - Governments and Privacy - Identify Theft - Privacy issues on the Web Data - Application of Cryptographic Techniques for Privacy Preservation.</td></tr>
<tr><td><b>Module:8</b></td><td colspan="2"><b>Contemporary Issues</b></td><td colspan="5"><b>2 hours</b></td></tr>
<tr><td></td><td colspan="2"><b>Total Lecture hours:</b></td><td colspan="5"><b>45 hours</b></td></tr>
</table>

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

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

Course Objectives
1. To impart the need of data privacy. 2. To categorize the statistical and computational techniques required to share data, with a primary focus on the social, and health sciences. 3. To formulate architectural, algorithmic, and technological foundations for the maintaining the data privacy.
Course Outcomes
After completion of this course, the student shall be able to: 1. Characterize basic rules, principles for protecting privacy and personally identifiable information. 2. Formulate data that supports useful statistical inference while minimizing the disclosure of sensitive information. 3. Identify the list of threats on the various types of anonymized data. 4. Classify and analyze the methods of test data generation with Privacy and utility.

Module:1	**Data privacy and Importance**	**5 hours**
Need for Sharing Data - Methods of Protecting Data - Importance of Balancing Data Privacy and Utility – Disclosure - Tabular Data - Micro data - Approaches to Statistical disclosure control – Ethics – principles - guidelines and regulations.		
Module:2	**Microdata**	**7 hours**
Disclosure - Disclosure risk - Estimating re-identification risk - Non-Perturbative Micro data masking - Perturbative Micro data masking - Information loss in Micro data.		
Module:3	**Static Data Anonymization on Multidimensional Data**	**7 hours**
Privacy – Preserving Methods - Classification of Data in a Multidimensional Dataset - Group-based Anonymization: k-Anonymity, l-Diversity, t-Closeness.		
Module:4	**Anonymization on Complex Data Structures**	**8 hours**
Privacy-Preserving Graph Data, Privacy-Preserving Time Series Data, Time Series Data Protection Methods, Privacy Preservation of Longitudinal Data, Privacy Preservation of Transaction Data.		
Module:5	**Threats to Anonymized Data**	**6 hours**
Threats to Anonymized Data, Threats to Data Structures, Threats by Anonymization Techniques: Randomization, k-Anonymization, l-Diversity, t-Closeness.		
Module:6	**Dynamic Data Protection**	**5 hours**
Dynamic Data Protection: Tokenization, Understanding Tokenization, Use Cases for Dynamic Data Protection, Benefits of Tokenization Compared to Other Methods, Components for Tokenization.		
Module:7	**Privacy-Preserving Test Data Generation and Privacy Regulations**	**5 hours**
Test Data Fundamentals - Insufficiencies of Anonymized Test Data. Privacy regulations: UK Data Protection Act, Swiss Data Protection Act, HIPPA, General Data Protection Regulation.		
Module:8	**Contemporary Issues**	**2 hours**
	Total Lecture hours:	**45 hours**

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

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

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

Course Objectives

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

Course Outcome

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

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

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

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

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

Course Objectives

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

Course Outcome

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

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

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

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

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

Mode of assessment: Continuous assessment / FAT

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

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

Course Objectives

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

Course Outcome

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

<table>
<tr><td colspan="6">Injection, and DOS - Securing Third-Party Dependencies.</td></tr>
<tr><td colspan="2"><b>Module:8</b></td><td colspan="2"><b>Contemporary Issues</b></td><td colspan="2"><b>2 hours</b></td></tr>
<tr><td colspan="6"></td></tr>
<tr><td colspan="2"></td><td colspan="2"><b>Total Lecture hours:</b></td><td colspan="2"><b>45 hours</b></td></tr>
<tr><td colspan="6"><b>Text Book</b></td></tr>
<tr><td>1.</td><td colspan="5">Andrew Hoffman, Web Application Security- Exploitation and Countermeasures for Modern Web Applications, March 2020, 1st Edition, O’Reilly Media, California.</td></tr>
<tr><td colspan="6"><b>Reference Books</b></td></tr>
<tr><td>1.</td><td colspan="5">D. Stuttard and M. Pinto, The Web Applications Hackers Handbook, 2011, 2nd Edition, Indianapolis, IN: Wiley, John Sons, United States.</td></tr>
<tr><td>2.</td><td colspan="5">Malcolm McDonald, Web Security for Developers: Real Threats, Practical Defense, 2020, Illustrated edition, No Starch Press, United States.</td></tr>
<tr><td colspan="6">Mode of Evaluation: CAT, Written Assignment, Quiz, FAT</td></tr>
<tr><td colspan="2">Recommended by Board of Studies</td><td colspan="4">04-03-2022</td></tr>
<tr><td colspan="2">Approved by Academic Council</td><td>No.65</td><td>Date</td><td colspan="2">17-03-2022</td></tr>
</table>

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

Course Objectives

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

Course Outcome

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

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

Module:1	Fundamentals of Malware Analysis	5 hours
Malware taxonomy - Malware analysis techniques – Packed and Obfuscated Malware - Portable Executable File Format: Headers and Sections, Malware Analysis in Virtual Machines - Malware Analysis Tools: ProcMon/ ProcExplore, BinText, FileAlyzer, OllyDbg, etc.		
Module:2	Static Analysis	4 hours
File signature analysis and Identifying file dependencies -Database of file hashes. String analysis - Local and online malware sandboxing - Levels of Abstraction - x86 Architecture - x86/x86_64 Assembly - Static Analysis Tools: PeiD, Dependency Walker, Resource Hacker.		
Module:3	Dynamic Analysis	4 hours
Source level vs. Assembly level Debuggers - Kernel vs. User-Mode Debugging – Exceptions - Modifying Execution with a Debugger - Modifying Program Execution in Practice - DLL analysis - Dynamic Analysis Tools: Virustotal, Malware Sandbox, Windows Sysinternals		
Module:4	Reverse Engineering	4 hours
Reverse engineering malicious code - Identifying malware passwords - Bypassing authentication -Advanced malware analysis: Virus, Trojan and APK Analysis - Reverse Engineering Tools: IDA Pro and OLLYDBG		
Module:5	Malicious Document Analysis	3 hours
PDF and Microsoft Office document structures – Identify PDF and office document vulnerabilities - Analysis of suspicious websites - Examining malicious documents: word, XL, PDF, and RTF files - Malware extraction and analysis tools.		
Module:6	Anti-Reverse-Engineering	3 hours
Anti-Disassembly - Anti-Debugging - Anti-Forensic Malware - Packers and Unpacking – Shellcode Analysis - 64-Bit Malware		
Module:7	Mobile Malware Analysis	5 hours
Mobile application penetration testing - Android and iOS Vulnerabilities - Exploit Prevention - Handheld Exploitation - Android Root Spreading and Distribution Android		

Debugging - Machine learning techniques for malware analysis: Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), Decision Trees (DT), Naïve Bayes (NB), and Neural Networks (NN).			
Module:8	**Contemporary Issues**	**2 hours**	
	Total Lecture hours:	**30 hours**	
Text Book			
1.	Abhijit Mohanta, Anoop Saldanha, Malware Analysis and Detection Engineering a Comprehensive Approach to Detect and Analyze Modern Malware, 2020, 1st edition, Apress (ISBN 978-1-4842-6192-7), United States.		
2.	M. Sikorski and A. Honig, Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software. 2012, 1st edition, No Starch Press San Francisco, CA. (ISBN No.: 9781593272906), United States.		
Reference Books			
1.	Monnappa K A, Learning Malware Analysis- Explore the concepts, tools, and techniques to analyze and investigate Windows malware, 2018, 1st edition, Packt Publishing, (ISBN 978-1-78839-250-1), United Kingdom.		
Mode of Evaluation: CAT / Assignment / Quiz / FAT / Seminar			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE321P	MALWARE ANALYSIS LAB	L	T	P	C
		0	0	2	1
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

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

Course Outcome

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

1. Apply techniques and concepts to unpack, extract, and decrypt malware.
2. Achieve proficiency with industry-standard malware analysis tools.

Indicative Experiments	
1	Examining PE Files using PEview, PE explorer and Resource Hacker • Disassembling Portable Executable (PE32) imports, exports, functions, main address, malicious string locations
2	Sandboxing malware using SANDBOX tool, Virus Total Analysis, Anyrun Analysis
3	Basic malware analysis: • file compilation date • imports/ exports, suspicious strings • run-time effect • procmon filter • hist -based signatures revealing files • registry keys, processes, services network-based signatures
4	Advanced static malware analysis • find address of main, code constructs, suspicious strings, • imported functions, their tasks, • intention of the malware impact of the malware via hex code
5	Analyze the malware using IDA Pro for reverse-engineering the malware: strings analysis, local variables, graph mode to cross-references, Analyzing Functions
6	Analyze the malware using OllyDbg: Debug the malware, Viewing Threads and Stacks, OllyDbg Code-Execution Options, Breakpoints, Loading DLLs, Exception Handling
7	Advanced analysis of Windows programs for processes, interactive remote shell, uploaded file, address of the subroutine, return value, Windows APIs
8	Malware behavior analysis • finding the source of malware • persistence mechanism, multiple instances replication mechanisms, hiding strategies • API calls for keylogging, constants involved post-infection actions of the malware, mutex, SendMessage API structure
9	Malware self-defense, packing and unpacking, obfuscation and de-obfuscation using Packers and obfuscation tools
10	Anti-disassembly and anti-debugging techniques used in the binary by patching the PE, set a breakpoint in the malicious subroutine
11	Analyzing malicious Microsoft Office and Adobe PDF documents to locate malicious

	embedded code such as shellcode, VBA macros or JavaScript, disassemble and/ or debug, shellcode analysis		
		Total Laboratory Hours	**30 hours**
Text Book(s)			
1.	M. Sikorski and A. Honig, Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software. 2012, 1st edition, No Starch Press San Francisco, CA. (ISBN No.: 9781593272906), United States.		
Reference Books			
1.	B. Dang, A. Gazet, E. Bachaalany, and S. Josse, Practical Reverse Engineering: X86, X64, arm, Windows Kernel, Reversing Tools, and Obfuscation. , 2014, Wiley, United States. (ISBN No. : 978-1-118-78731-1)		
Mode of assessment: Continuous assessment / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE322L	DIGITAL FORENSICS	L	T	P	C
		2	0	0	2
Pre-requisite	NIL	Syllabus version			
		1.0			

Course Objectives

1. To present a comprehensive perception of digital forensic principles, collection, preservation, and analysis of digital evidence.
2. To enlighten the importance of forensic procedures, legal considerations, digital evidence controls, and the documentation of forensic analysis.
3. To develop a comprehension of the different tools and methods for conducting digital forensic acquisition and analysis.

Course Outcomes

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

1. Understand the responsibilities and liabilities of a computer forensic investigator
2. Seize a computer from a crime scene without damage and follow the legal procedures and standards.
3. Demonstrate the ability to perform forensic data acquisition and analysis.
4. Analyze and retrieve hidden and damaged files from different operating systems.
5. Apply forensics to recent technologies such as smart phones, email, cloud and social media.

Module:1	Understanding Digital Forensics and Legal Aspects	3 hours
Understanding computer forensics - Preparing for computer investigation – Maintaining professional conduct – understanding computer investigations – Taking a systematic approach – Corporate Hi-Tech investigations – Conducting an investigation.		
Module:2	**Acquisition and Storage of Data**	**4 hours**
Understanding Storage Formats for Digital Evidence - Determining the Best Acquisition Method - Contingency Planning for Image Acquisitions - Using Acquisition Tools - Validating Data Acquisitions - Performing RAID Data Acquisitions - Using Remote Network Acquisition Tools - Storing Digital Evidence - Obtaining a Digital Hash - Sample Cases.		
Module:3	**Working with Windows**	**5 hours**
Understanding File Systems - Exploring Microsoft File Structures - Examining NTFS Disks - Understanding Whole Disk Encryption - Understanding the Windows Registry - Understanding Microsoft Startup Tasks - Understanding MS-DOS Startup Tasks - Evaluating Computer Forensics Tool Needs - Computer Forensics Software and Hardware Tools.		
Module:4	**Working with Linux/Unix Systems**	**4 hours**
UNIX and Linux Overview - Inodes - Boot Process - Drives and Partition Schemes - Examining disk Structures - Understanding Other Disk Structures - Ownership and Permissions, File Attributes, Hidden Files, User Accounts - Case studies - Validating Forensic Data – Addressing Data-Hiding Techniques – Locating and Recovering Graphics File.		
Module:5	**Email and Social Media Forensics**	**4 hours**
Investigating E-mail crimes and Violations – Applying Digital Forensics Methods to Social Media Communications - Social Media Forensics on Mobile Devices - Forensics Tools for Social Media Investigations.		
Module:6	**Mobile Forensics**	**4 hours**
Mobile phone basics – Acquisition procedures for mobile - Android Device –Android Malware – SIM Forensic Analysis – Case study.		
Module:7	**Cloud Forensics**	**4 hours**

Working with the cloud vendor, obtaining evidence, reviewing logs and APIs.			
Module:8	**Contemporary Issues**		**2 hours**
	Total Lecture hours:		**30 hours**
Text Book(s)			
1.	B. Nelson, A. Phillips, F. Enfinger, and C. Steuart, Guide to Computer Forensics and Investigations, 2019, 6th ed. CENGAGE, INDIA (ISBN: 9789353506261)		
Reference Books			
1.	André Årnes, Digital Forensics, 2018, 1st ed., Wiley, USA(ISBN No.: 9781119262411)		
2.	Nihad A Hassan, Digital Forensics Basics: A Practical Guide to Using Windows OS, 2019, 1st ed, APress, USA (ISBN: 9781484238387)		
Mode of Evaluation: CAT, assignment, Quiz and FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022

BCSE322P	**DIGITAL FORENSICS LAB**	**L**	**T**	**P**	**C**
		0	**0**	**2**	**1**
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To present a comprehensive perception of digital forensic principles, collection, preservation, and analysis of digital evidence.
2. To enlighten the importance of forensic procedures, legal considerations, digital evidence controls, and the documentation of forensic analysis.
3. To develop a comprehension of the different tools and methods for conducting digital forensic acquisition and analysis.

Course Outcomes

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

1. Demonstrate the ability to perform forensic data acquisition and analysis.
2. Apply forensics to recent technologies such as smart phones, email, cloud and social media.

Indicative Experiments

1.	Extract the features based on various color models and apply on image and video retrieval
2.	File Recovery (Deleted, fragmented, hidden)
3.	Network Forensics (Determining the type attacks, extracting files from network logs, encrypted _les)
4.	OS Forensics (Windows and Linux artifacts, memory, registry)
5.	Mobile Forensics(Tools for Android and iOS)
6.	Mobile Forensics(Tools for Android and iOS)
7.	Social Media Forensics
Total Laboratory Hours	**30 hours**

Text Book

1.	B. Nelson, A. Phillips, F. Enfinger, and C. Steuart, Guide to Computer Forensics and Investigations, 2019, 6th ed. CENGAGE, INDIA (ISBN: 9789353506261)

Reference Books

1.	Nihad A Hassan, Digital Forensics Basics: A Practical Guide to Using Windows OS, 2019, 1st ed, APress, USA (ISBN: 9781484238387)

Mode of assessment: Continuous assessment / FAT

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

BCSE323L	**DIGITAL WATERMARKING AND STEGANOGRAPHY**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**
Pre-requisite	**NIL**	**Syllabus version**			
		1.0			

Course Objectives

1. To understand the basic principles, characteristics, various approaches and applications of digital watermarking and steganography.
2. To apply digital watermarking techniques as an authentication tool for distribution of content over the Internet and steganography techniques for covert communication.
3. To impart knowledge on the basics of the counter measures like steganalysis for assessing the data hiding methods.

Course Outcome

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

1. Learn the fundamental concepts, principles, characteristics and performance measures of digital watermarking and steganography.
2. Acquire the various concepts of watermarking for digital authentication and authorization schemes related to electronic documents, image and video.
3. Gathering the various concepts of steganography to access the sensitive information concealing of message, image, audio or video within another file.
4. Design and implement efficient data hiding methods against steganalysis techniques.

Module:1	**Fundamentals of Digital Watermarking**	**6 hours**

Importance of Watermarking - Application and Properties of Watermarking - Models of Watermarking - Basic Message Coding: Mapping Message into Message Vectors, Error Correction Coding - Watermarking with Side Information - Analyzing Errors.

Module:2	**Digital Watermarking Schemes**	**7 hours**

Spatial Domain: Correlation based Watermarking, Least Significant bit Watermarking - Frequency domain: Discrete Wavelet Transform Watermarking, Discrete Fourier Transform Watermarking, Discrete Cosine Watermarking, Quantization Watermarking, Haar Transform Watermarking, Hadamard Transform Watermarking - Robust Watermarking - Fragile and Semi Fragile Watermarking.

Module:3	**Digital Watermarking Security and Authentication**	**5 hours**

Watermarking Security: Security Requirements, Watermark Security and Cryptography, Watermarking Attacks and Tools - Content Authentication: Exact Authentication, Selective Authentication, Localization, Restoration.

Module:4	**Steganography**	**7 hours**

Basics and Importance of Steganography - Applications and Properties of Steganography - Steganography: LSB embedding, Steganography in palette images -Steganography in JPEG images: JSteg data hiding in spatial and transform domain -Steganography Security.

Module:5	**Audio and Video Steganography**	**6 hours**

Audio Steganography: Temporal domain techniques, Transform domain techniques, Cepstral Domain - Video Steganography: Introduction Video Streams, Substitution-Based Techniques, Transform Domain Techniques, Adaptive Techniques, Format-Based Techniques - Cover Generation Techniques Video Quality Metrics - Perceptual Transparency Analysis - Robustness against Compression and Manipulation.

Module:6	**Wet Paper Code**	**6 hours**

Random Linear Codes - LT Codes - Perturbed Quantization, Matrix Embedding - Matrix Embedding Theorem - Binary Hamming Codes - Q-Ary Case Random Linear Codes for Large Payloads.

Module:7	**Steganalysis**	**6 hours**

Steganalysis Principles - Statistical Steganalysis: Steganalysis as detection problem,

Modeling images using features, Receiver operating Characteristics - Targeted Steganalysis : Sample pair analysis, Targeted attack on F5 using Calibration, Targeted attack on ± embedding - Blind Steganalysis: Features for steganalysis of JPEG images (cover vs all-stego and one class neighbor machine).			
Module:8	**Contemporary Isues**		**2 hours**
	Total Lecture hours:		**45 hours**
Text Book(s)			
1.	Frank Y. Shih, Digital Watermarking and Steganography Fundamentals and Techniques, 2020, 2nd Ed. CRC Press, United States. (ISBN No. : 9780367656430)		
2.	J. Fridrich, Steganography in Digital Media: Principles, Algorithms, and Applications, 2010, 1st Ed. Cambridge: Cambridge University Press, United Kingdom. (ISBN No.: 978-0-52-119019-0)		
Reference Books			
1.	I. J. Cox, M. L. Miller, J. A. Bloom, T. Kalker, and J. Fridrich, Digital Watermarking and Steganography, 2008, 2nd Ed. Amsterdam: Morgan Kaufmann Publishers In, United States. (ISBN No. : 978-0-12-372585-1)		
2.	P. Wayner, Disappearing Cryptography: Information hiding: Steganography and Watermarking, 2008, 3rd ed. Amsterdam: Morgan Kaufmann Publishers In, United States. (ISBN No. : 978-0-08-092270-6)		
Mode of Evaluation: CAT / Assignment / Quiz / FAT			
Recommended by Board of Studies	04-03-2022		
Approved by Academic Council	No.65	Date	17-03-2022