SYLLABUS
FOR
BACHELOR OF TECHNOLOGY
IN COMPUTER SCIENCE AND ENGINEERING
(CYBER SECURITY)

SCHOOL OF ENGINEERING
UNIVERSITY OF KASHMIR
SRINAGR

JULY -2025

(Applicable to Batch 2025 & Onwards)

Office of the Dean, School of Engineering

University of Kashmir, Srinagar

(NAAC Accredited Grade- A++)

Srinagar-190006 J&K (India)

...ce of the Board of Studies (BoS) members at the meeting held on **30.06.2026** in connection with ...raming/revision of the syllabus for the following B.Tech. programmes offered at the Institute of Technology, Zakura Campus; North Campus, Delina; SSM College of Engineering; and Kashmir College of Engineering & Technology.

Board of Studies for B.Tech. programmes i) Computer Science and Engineering (Artificial Intelligence) ii) Computer Science and Engineering (Cyber Security) iii) Artificial Intelligence & Data Science iv) Computer Science and Engineering (CSE), v) Artificial Intelligence and vi) Computer Engineering.			***Signatures***
1	Dean, School of Engineering University of Kashmir	Chairman	
2	Director, Institute of Technology Zakura Campus	Member	
3	Prof. S.M.A.Andrabi Head, Dept. of Applied Sciences, IOT, Zakura Campus	Member	
4	Director, North Campus	Member	30/6/26
5	Head, Dept. of Computer Sciences University of Kashmir	Member	30.6.26
6	Head, Dept. of Computer Science and Engineering, NIT, Sgr	Member	
7	Coordinator Dept of Computer Science and Engineering, IOT Zakura Campus	Member	
8	Coordinator Dept. of Computer Science and Engineering, NG	Member	30/06/26.
9	Director, NIELIT, Rangreth, Srinagar	Member	
10	Dr. Manzoor Ahmad Chachoo Dept. of Computer Science, University of Kashmir	Member	30/6/2026
11	Dr. Suhail Ahmad Mir Dept. of Computer Science and Engineering, NC	Member	30/6/26.
12	Er. Khalid Hussain Dept. of Computer Science and Engineering, NC	Member	
13	Dr. Md. Hesam Akhter Dept. of Computer Science and Engineering, NC	Member	30/6/26
14	Head, Dept. of Computer Science Engineering, SSM College of Engineering	Member	30/06/26

Office of the Dean, School of Engineering

University of Kashmir, Srinagar

(NAAC Accredited Grade- A++)

Srinagar-190006 J&K (India)

Asifa Bashir DR. OMAR.

15	Head, Dept. of Computer Engineering/AI Kashmir College of Engineering & Technology	Member		

B. Tech Computer Science and Engineering (Cyber Security)	
Programme Specific Outcomes	
PSO No.	**Programme Specific Outcomes**
PSO1	Apply Computer Science fundamentals and Cyber Security principles to solve real-world computing problems.
PSO2	Design and develop secure software, networks, and computing systems using modern technologies.
PSO3	Analyse and mitigate cyber threats using cryptography, ethical hacking, digital forensics, and security tools.
PS04	Apply emerging technologies such as Artificial Intelligence, Cloud Computing, IoT, and Blockchain for secure computing solutions.
PSO5	Demonstrate professional ethics, legal compliance, innovation, and lifelong learning in Computer Science and Cyber Security.
Programme Learning Outcomes	
PLO1	**Engineering Knowledge:** Apply knowledge of mathematics, science, and core computer science concepts to solve complex engineering problems.
PLO2	**Problem Analysis:** Identify, analyse, and formulate computing problems to reach logical conclusions using domain knowledge and appropriate algorithms.
PLO3	**Design/Development of Solutions:** Design and evaluate computing systems, components, or algorithms that meet desired specifications under realistic constraints.
PLO4	**Investigation of Complex Problems:** Conduct experiments, interpret data, and apply research methods to draw valid conclusions in computing and software systems.
PLO5	**Modern Tool Usage:** Use appropriate modern tools and technologies, including development environments, software frameworks, and debugging tools, for software development.
PLO6	**The Engineer and Society:** Apply contextual knowledge to assess societal, health, safety, legal, and cultural issues relevant to computing technologies.
PLO7	**Environment and Sustainability:** Understand and apply computing solutions that promote environmental sustainability and energy-efficient practices.
PLO8	**Ethics:** Commit to ethical principles and legal responsibilities in the design, development, and deployment of computing systems.
PLO9	**Individual and Team Work:** Function effectively as an individual, or as a member or leader in diverse, multidisciplinary teams.
PLO10	**Communication:** Communicate effectively on technical topics with the computing community and society through presentations, documentation, and instructions.
PLO11	**Project Management and Finance:** Apply software project management principles and financial understanding to lead or participate in computing project execution.
PLO12	**Innovation, Entrepreneurship and Life-long Learning:** Exhibit an innovative mindset and entrepreneurial skills for technology development or engage in lifelong learning to adapt to emerging computing trends, tools, and practices throughout one's career.
Accreditation Alignment	
1. The programme is designed in accordance with NEP 2020, AICTE, and NBA Outcome-Based Education (OBE) guidelines. 2. Programme Learning Outcomes (PLOs) align with the Washington Accord Graduate Attributes for Engineering Graduates. 3. The curriculum integrates GATE, UPSC, industry requirements, research, innovation, and emerging Cyber Security technologies to prepare graduates for higher studies, research, entrepreneurship, and professional careers.	

Course Code Formula

Position:	1	2	3	4	5	6	7	8	9	10
Indicator:	**B**	**C**	**C**	**S**	**B**	**M**	**T**	**3**	**2**	**5**

Digit	**Description**
1	Bachelor's Programme
2-4	Programme Code: Computer Science and Engineering (Cyber Secur) = CCS
5	Indicator Alphabet in Course Code
6-7	Course Title
8	Semester (1 to 8)
9-10	Year of Launch

Indicator Alphabet	**Description**
H	Humanities & Social Science Course
B	Basic Science Course
E	Engineering Science Course
C	Programme Core Course
D	Programme Elective Course
O	Open Elective Course
L	Laboratory Course
P	Project/internship
Y	Seminar
A	Audit Course

Examination Code	**Description**
MSE	Mid Semester Evaluation
IA	Internal Assessment
CIE	Continuous Internal Evaluation = MSE + IA
SEE	Semester End Evaluation

Examination Pattern

SEMESTER END EVALUATION(SEE)			
Semester-end Examination (Paper Pattern)			
Section	**No of questions**	**Marks**	**Sectional Marks**
A	10 (2 from each unit)	1	10
B	5 (1 from each unit)	4	20
C	2 out of 5 (1 from each unit)	10	20
Total			50

CONTINUOUS INTERNAL EVALUATION(CIE)			
Mid Semester Evaluation (MSE)			
Mid-Term Examination (Paper Pattern)			
Section	**No of questions**	**Marks**	**Sectional Marks**
A	10 (5 from each unit)	1	10
B	3	5	15
C	1 out of 2 (1 from each unit 1 & 2)	10	10
Total			35

INTERNAL ASSESSMENT			
Section	**Type of Assessment**	**Marks**	**Sectional Marks**
1	Attendance Viva Quiz Presentation Surprise test Mini Project, etc	The Teacher can divide the marks across the assessments	15
Total			15

Course Structure, B.Tech Computer Science and Engineering (Cyber Security)

3rd Semester to 8th Semester

Semester: - III								Examination Scheme (Distribution of Marks)										
								CIE (Theory)			CIE (Practical)							
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSCDM326	Discrete Mathematics	2	1	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
2	BCCSCOO326	Object Oriented Programming with Java	2	0	4	4	6	35	15	50	35	15	50	**100**	50	50	**100**	**200**
3	BCCSCDB326	Database Management System	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
4	BCCSCDD326	Digital System Design	2	1	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
5	BCCSCSS326	Signals and Systems	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
6	BCCSLCP326	Professional Practice and Digital Skills Laboratory	0	0	4	2	4	0	0	0	35	15	50	**50**	0	0	**0**	**50**
Total Hours			**13**	**1**	**12**	**20**	**25**											

Semester: - IV								Examination Scheme (Distribution of Marks)										
								CIE (Theory)			CIE (Practical)							
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSCPS426	Probability Statistics and Queuing	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
2	BCCSCDS426	Data Structures using C	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
3	BCCSCMP426	Microprocessors	2	1	2	4	4	35	15	50	35	15	50	**100**	50	50	**100**	**200**
4	BCCSCAD426	Analysis and Design of Algorithms	2	0	4	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
5	BCCSCCS426	Communication Systems	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
6	BCCSLPP426	Python Programming	0	0	4	2	4	0	0	0	35	15	50	**50**	0	0	**0**	**50**
Total Hours			**14**	**0**	**12**	**20**	**24**											

Semester: - V								Examination Scheme (Distribution of Marks)									
								CIE (Theory)			CIE (Practical)						
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSCOS526	Operating Systems	2	1	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
2	BCCSCAM526	Artificial Intelligence and Machine Learning	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
3	BCCSCCA526	Computer Organization and Architecture	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
4	BCCSCTC526	Theory of Computation	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
5	BCCSCCN526	Computer Networks	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
6	BCCSLIP526	IoT & Smart System Prototyping	0	0	4	2	4	0	0	0	35	15	50	**50**	0	0	**0**	**50**
Total Hours			**14**	**1**	**10**	**20**	**25**											

Semester: - VI								Examination Scheme (Distribution of Marks)									
								CIE (Theory)			CIE (Practical)						
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSCCN626	Cryptography & Network Security	3	0	2	4	3	35	15	50	35	15	50	**100**	50	50	**100**	**200**
2	BCCSCWM626	Wireless and mobile Network security	3	1	0	4	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
3	BCCSCSD626	Secure Software Development	2	0	2	3	3	35	15	50	35	15	50	**100**	50	50	**100**	**200**
4	BCCSCEH626	Ethical Hacking	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
5	BCCSLWP626	Web programming	0	0	4	2	4	0	0	0	35	15	50	**50**	0	0	**0**	**50**
6	BCCSDXX626	Elective-I	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
Total Hours			**14**	**1**	**10**	**20**	**21**											

Semester: - VII								Examination Scheme (Distribution of Marks)									
								CIE (Theory)			CIE (Practical)						
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSCCC726	Cloud Computing	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
2	BCCSLIE726	Innovation, Entrepreneurship & Startups	0	0	4	2	4	0	0	0	35	15	50	**50**	0	0	**0**	**50**
3	BCCSDXX726	Elective-II	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
4	BCCSDXX726	Elective-III	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
5	BCCSOXX726	Open Elective-I	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
6	BCCSPMP726	Project-I	0	0	6	3	6	0	0	0	70	30	100	**100**	0	100	**100**	**200**
Total Hours			**12**	**0**	**16**	**20**	**28**											

Semester: - VIII								Examination Scheme (Distribution of Marks)									
								CIE (Theory)			CIE (Practical)						
S. No.	Course Code	Course Title	L	T	P	Credits	Hours	MSE	CA	Total	MSE	CA	Total	Internal (Total)	ESA (Theory)	ESA (Practical)	External (Total)	Total Marks
1	BCCSDXX826	Elective-IV	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
2	BCCSDXX826	Elective-V	3	0	2	4	5	35	15	50	35	15	50	**100**	50	50	**100**	**200**
3	BCCSOXX826	Open Elective-II	3	0	0	3	3	35	15	50	0	0	0	**50**	50	0	**50**	**100**
4	BCCSPMP826	Project-II	0	0	18	9	18	0	0	0	70	30	100	**100**	0	200	**200**	**300**
Total Hours			**9**	**0**	**22**	**20**	**31**											

List of Electives

S. No.	Course Code	Elective-I
01	BCCSDBT626	Blockchain Technology
02	BCCSDCF626	Cyber Forensics
03	BCCSDSS626	System Security

S. No.	Course Code	Elective-II
01	BCCSDVP726	Vulnerability Assessment and Penetration Testing
02	BCCSDMR726	Malware Analysis and Reverse Engineering
03	BCCSDCT726	Cyber Threat Intelligence

S. No.	Course Code	Elective-III
01	BCCSDAC726	Artificial Intelligence for Cybersecurity
02	BCCSDCL726	Privacy, Security, and Cyber Law
03	BCCSDSD726	DevSecOps and Secure Deployment

S. No.	Course Code	Open Elective-I
01	BCCSOES726	Intelligent E-commerce Systems
02	BCCSOMA726	Multimedia and Animations
03	BCCSOPP726	Python Programming for Engineers
04	BCCSOCS726	Fundamentals of Cyber Security

S. No.	Course Code	Elective-IV
01	BCCSDBD826	Big Data Analytics for Cyber Security
02	BCCSDCG826	Cyber Governance, Warfare and National Security
03	BCCSDDW826	Digital Watermarking and Steganography

S. No.	Course Code	Elective-V
01	BCCSDIT826	Internet of Things and Security
02	BCCSDHP826	Parallel and High-Performance Computing
03	BCCSDQC826	Fundamentals of quantum Computing

S. No.	Course Code	Open Elective-II
01	BCCSOBA826	Blockchain Technology and Application
02	BCCSOAS826	AI for Startups
03	BCCSOFA826	Fundamentals of AI and DS
04	BCCSOPE826	Prompt Engineering

3rd Semester

Course Code	BCCSCDM326				3rd Semester	
Course Title	Discrete Mathematics				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	Nil

Course Learning Outcomes (CLOs)	
CLO1	Apply principles of mathematical logic, proof techniques, sets and relations to solve computational problems.
CLO2	Analyze functions, counting techniques, permutations, combinations and recurrence relations for algorithmic problem solving
CLO3	Model and solve real-world problems using graph theory, trees and graph traversal algorithms.
CLO4	Explain algebraic structures including groups, rings, fields and lattices, and relate them to computing concepts.
CLO5	Apply discrete mathematical concepts in computer science applications such as Boolean logic, cryptography, Artificial Intelligence, machine learning, data mining and computer networks

Syllabus

Units	Content
1	**Mathematical Logic and Proof Techniques:** Propositions, Logical Connectives, Truth Tables, Tautologies, Contradictions, Logical Equivalence, Predicate Logic, Quantifiers, Rules of Inference, Normal Forms (CNF and DNF), Methods of Proof, Direct Proof, Proof by Contradiction, Contrapositive Method, Mathematical Induction, and Strong Induction.Sets and Set Operations, Cartesian Product, Relations, Equivalence Relations, Partial Order Relations, Hasse Diagrams
2	**Functions, Counting Principles, and Recurrence Relations** Functions, types of functions, one-to-one and onto functions, composition and inverse of functions, recursive functions, fundamental principles of counting, permutations and combinations, binomial theorem, principle of inclusion and exclusion, pigeonhole principle, recurrence relations, homogeneous and non-homogeneous linear recurrence relations with constant coefficients and their solutions.

3	**Graph Theory and Trees** Introduction to graphs, graph terminology, types of graphs, graph representation using adjacency matrix and adjacency list, degree of vertices, walks, trails, paths, circuits, connected graphs, bipartite graphs, Euler graphs, Hamiltonian graphs, graph isomorphism, trees and their properties, rooted trees, binary trees, spanning trees, minimum spanning trees, graph traversal techniques including Breadth First Search (BFS) and Depth First Search (DFS).
4	**Algebraic Structures and Lattices** Binary operations, semigroups, monoids, groups, subgroups, cyclic groups, permutation groups, Lagrange's theorem, cosets, normal subgroups, group homomorphism, introduction to rings and fields, partially ordered sets (Posets), lattices, bounded lattices, distributive lattices, and Boolean lattice
5	**Applications of Discrete Mathematics** Boolean algebra, Boolean functions, switching algebra, finite state machines, graph algorithms, applications of graph theory in computer networks, mathematical foundations of cryptography, discrete structures in Artificial Intelligence, knowledge representation, search techniques, data mining, machine learning and cyber security applications.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	3	2	1	0	0	0	0	0	0	0	1	2.0
CLO2	3	3	3	2	0	0	0	0	0	0	0	1	2.4
CLO3	2	3	3	2	1	0	0	0	0	0	0	2	2.2
CLO4	3	2	2	1	0	0	0	0	0	0	0	2	2.0
CLO5	3	3	3	2	2	0	0	0	0	0	0	3	2.7
Avg PLO	2.8	2.8	2.6	1.6	1.5	0	0	0	0	0	0	1.8	2.26

Suggested Reading

1	*Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw-Hill.*
2	*Ralph P. Grimaldi, Discrete and Combinatorial Mathematics, Pearson.*
3	*J. P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, McGraw-Hill.*
4	*Susanna S. Epp, Discrete Mathematics with Applications, Cengage Learning.*

5	*Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science, PHI Learning.*

Teaching-Learning Strategies

The course will be delivered through interactive lectures, tutorial sessions, problem-solving exercises, board work, multimedia presentations, case studies, algorithm demonstrations, collaborative learning activities, assignments, quizzes and application-oriented discussions using suitable software tools wherever applicable development.

Course Code	BCCSCDB326				3rd Semester	
Course Title	Database Management System				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	2	4	100	100

Course Learning Outcomes (CLOs)	
CLO1	Explain the architecture, components, data models and design principles of modern Database Management Systems.
CLO2	Design relational database schemas using Entity-Relationship (ER) modeling, normalization techniques and Structured Query Language (SQL).
CLO3	Develop database applications by implementing SQL queries, constraints, stored procedures, triggers and transaction management techniques.
CLO4	Apply indexing, concurrency control, recovery mechanisms and database security techniques to ensure efficient and reliable database operations
CLO5	Evaluate and utilize modern database technologies including distributed databases, NoSQL systems, cloud databases and their applications in Artificial Intelligence, Cyber Security, Data Analytics and Internet of Things.

Syllabus

Units	Content
1	**Introduction to Database Systems and Data Modeling** Introduction to database systems, characteristics and advantages of DBMS, database users, database administrator (DBA), database architecture, three-schema architecture, data independence, data models, schema and instances, data dictionary, database languages, Entity-Relationship (ER) model, entities, attributes, keys, relationships, mapping cardinalities, participation constraints, enhanced ER model, specialization, generalization, aggregation, and ER-to-relational mapping.
2	**Relational Database Model and SQL** Relational data model, integrity constraints, relational algebra, relational calculus (overview), Structured Query Language (SQL), data definition language (DDL), data manipulation language (DML), data control language (DCL), transaction control language (TCL), joins, nested queries, aggregate functions, views, indexes, sequences, constraints, stored procedures, functions and triggers.

3	**Database Design and Transaction Management** Functional dependencies, inference rules, closure of attributes, normalization, first, second, third and Boyce-Codd normal forms (1NF, 2NF, 3NF and BCNF), multivalued dependencies, fourth normal form (4NF), fifth normal form (5NF), lossless decomposition, dependency preservation, transaction concepts, ACID properties, transaction states, serializability, concurrency control, locking protocols, timestamp-based protocols, deadlocks and deadlock handling
4	**Storage Organization, Recovery and Database Security** File organization, storage structures, indexing techniques, primary and secondary indexes, clustered and non-clustered indexes, B+ trees, hashing techniques, query processing, query optimization (introduction), recovery techniques, log-based recovery, checkpoints, shadow paging, database backup and recovery, database security, authentication, authorization, role-based access control, SQL injection, data integrity, auditing and privacy preservation.
5	**Advanced Database Systems** Distributed databases, distributed transactions, database fragmentation and replication, parallel databases, NoSQL databases, key-value, document, column-family and graph databases, MongoDB fundamentals, data warehousing, data mining concepts, cloud databases, big data storage, database applications in Artificial Intelligence, Machine Learning, Cyber Security and Internet of Things.
S. No.	**Suggested Practicals / Mini Projects**
1	Installation and familiarization with Database Management Systems such as MySQL/PostgreSQL and introduction to database administration tools.
2	Creation of databases and tables using SQL Data Definition Language (DDL) commands including CREATE, ALTER, DROP, TRUNCATE and RENAME along with appropriate data types.
3	Implementation of integrity constraints including PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK and DEFAULT constraints.
4	Execution of Data Manipulation Language (DML) commands including INSERT, UPDATE, DELETE and SELECT with conditional retrieval, sorting, pattern matching and NULL value handling.
5	Implementation of SQL operators, aggregate functions, GROUP BY, HAVING clause and ORDER BY clause.
6	Execution of different types of SQL joins including INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTER JOIN, SELF JOIN and CROSS JOIN along with nested queries and correlated subqueries
7	Creation and manipulation of database objects including Views, Indexes and Sequences, and implementation of Data Control Language (DCL) commands such as GRANT and REVOKE
8	Implementation of Stored Procedures, User Defined Functions, Triggers and Cursors for database automation.
9	Design of Entity Relationship (ER) diagrams for real-world applications and conversion of ER models into relational database schemas.

10	Functional Dependency analysis and normalization of relational schemas up to BCNF using suitable case studies.
11	Demonstration of transaction processing using COMMIT, ROLLBACK and SAVEPOINT commands, and study of ACID properties and concurrent transaction execution.
12	Introduction to NoSQL databases using MongoDB including database creation, CRUD operations, document modeling and basic aggregation queries.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	2	1	2	0	0	0	0	0	0	1	1.83
CLO2	3	3	3	2	2	0	0	0	0	0	0	1	2.33
CLO3	2	3	3	2	3	0	0	0	0	0	0	2	2.50
CLO4	2	3	3	3	3	0	0	2	0	0	0	2	2.57
CLO5	2	2	3	2	3	0	0	0	0	0	0	3	2.50
Avg PLO	2.40	2.60	2.80	2.00	2.60	0	0	2.0	0	0	0	1.80	2.35

Suggested Reading

1	*R. Elmasri and S. B. Navathe, Fundamentals of Database Systems, Pearson.*
2	*Abraham Silberschatz, Henry F. Korth and S. Sudarshan, Database System Concepts, McGraw-Hill.*
3	*Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, McGraw-Hill.*
4	*C. J. Date, An Introduction to Database Systems, Pearson.*
5	*Pramod J. Sadalage and Martin Fowler, NoSQL Distilled, Addison-Wesley.*

Teaching-Learning Strategies

The course will be delivered through interactive classroom lectures, multimedia presentations, tutorial sessions, hands-on laboratory experiments, SQL programming exercises, database design case studies, problem-solving sessions, assignments, quizzes, mini-projects, seminars, collaborative learning activities and industry-oriented demonstrations using modern DBMS tools such as MySQL, PostgreSQL and MongoDB.

Course Code	BCCSCDD326				3rd Semester	
Course Title	Digital Systems Design				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	2	1	2	4	100	100

Course Learning Outcomes (CLOs)	
CLO1	Explain the fundamental concepts of digital systems, number systems, Boolean algebra, and logic gates.
CLO2	Analyze and design combinational logic circuits for digital system applications.
CLO3	Analyze the operation and design of sequential logic circuits using flip-flops and related timing concepts.
CLO4	Apply sequential circuit design principles to registers, counters, memory organization, and error control coding.
CLO5	Design, implement, and simulate digital systems using VHDL and Hardware Description Language (HDL) design methodologies.

Syllabus

Units	Content
1	**Number Systems and Logic Gates:** Digital systems and binary logic, number systems, number system conversions, binary arithmetic, signed number representation, Boolean algebra, logic gates, universal gates, logic families (TTL and CMOS – overview), Karnaugh maps (K-map) for logic simplification.
2	**Combinational Logic Circuits and Error Control Coding:** Design and analysis of combinational circuits, adders, subtractors, multiplexers, demultiplexers, encoders, decoder, parity generation and checking, Hamming code, error detection and correction, Programmable Logic Arrays (PLA) and Programmable Array Logic (PAL) (concept).
3	**Sequential Logic Fundamentals:** Latches, flip-flops (SR, JK, D, T), characteristic tables, excitation tables, characteristic equations, flip-flop conversions, race-around condition, setup time and hold time, timing diagrams.

4	**Sequential Circuit Design and Memory Organization:** Registers, shift registers, synchronous and asynchronous counters, finite state machine (FSM) concepts, memory organization, memory cells, memory arrays, memory addressing, read and write operations, semiconductor memories (RAM and ROM).
5	**Hardware Description Language and Digital System Design using VHDL:** Introduction to Hardware Description Languages, VHDL design flow, VHDL program structure, libraries, entity, architecture, data types, operators, signals and variables, concurrent and sequential statements, behavioral, dataflow and structural modeling, VHDL implementation of combinational and sequential circuits, simulation and synthesis (concept).
S. No.	**Suggested Practicals / Mini Projects**
1	Familiarization with the Digital Electronics Laboratory and study of electronic measuring instruments, including digital multimeter, regulated DC power supply, function generator, logic probe, logic analyzer (demonstration), and Digital Storage Oscilloscope (DS
2	Measurement and analysis of digital waveforms using a Digital Storage Oscilloscope (DSO), verification of logic levels, propagation delay, pulse width, frequency, and timing characteristics of digital signals.
3	Verification of basic logic gates and universal gates using standard digital ICs (74xx/40xx series) and realization of Boolean functions.
4	Design and implementation of combinational logic circuits using standard digital ICs, including adders, subtractors, multiplexers, demultiplexers, encoders, decoders, and comparators.
5	Design and implementation of a BCD-to-7-segment display decoder using standard digital ICs and verification of its operation.
6	Design and implementation of flip-flops, shift registers, and synchronous/asynchronous counters using standard digital ICs.
7	Introduction to the VHDL programming environment and implementation of basic logic gates and combinational circuits using VHDL.
8	Design and implementation of combinational and sequential circuits using VHDL
9	Design and simulation of finite state machines (FSMs) using VHDL.
10	Mini Project: Design, implement, and simulate a digital system such as a traffic light controller, digital clock, sequence detector etc. using VHDL.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	1	0	1	0	0	0	0	0	0	1	1.60
CLO2	3	3	3	2	2	0	0	0	0	0	0	1	2.33
CLO3	3	3	3	2	2	0	0	0	0	0	0	1	2.33

CLO4	3	3	3	2	2	0	0	0	0	0	0	2	2.50
CLO5	2	3	3	2	3	0	0	0	0	0	0	3	2.67
Avg PLO	2.80	2.80	2.60	2.00	2.00	0	0	0	0	0	0	1.60	2.29

Suggested Reading

1	*M. Morris Mano and Michael D. Ciletti, Digital Design, Pearson.*
2	*Ronald J. Tocci, Neal S. Widmer and Gregory L. Moss, Digital Systems: Principles and Applications, Pearson.*
3	*John F. Wakerly, Digital Design: Principles and Practices, Pearson.*
4	*A. Anand Kumar, Fundamentals of Digital Circuits, PHI Learning.*
5	*Charles H. Roth Jr. and Larry L. Kinney, Fundamentals of Logic Design, Cengage Learning.*

Teaching-Learning Strategies

Interactive lectures supported by numerical examples, logic design problems, and demonstrations to explain the principles of digital electronics and digital system design.

Hands-on laboratory sessions involving digital ICs, electronic measuring instruments, and VHDL-based design and simulation of combinational and sequential circuits.

Problem-solving tutorials and assignments on Boolean algebra, logic simplification, combinational and sequential circuit design, memory organization, and error control coding.

Project-based learning involves the design, implementation, and simulation of digital systems using VHDL and appropriate digital design tools.

Course Code	BCCSCOO326				3rd Semester	
Course Title	Object Oriented Programming with Java				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	2	0	4	4	100	100

Course Learning Outcomes (CLOs)

CLO1	Demonstrate proficiency in Java programming fundamentals, including program structure, data types, control statements, arrays, classes, objects, methods, constructors, and access modifiers, to develop object-oriented applications.
CLO2	Apply object-oriented programming principles such as inheritance, polymorphism, abstraction, interfaces, packages, and dynamic binding to design and implement modular, reusable, and maintainable Java applications.
CLO3	Develop reliable and concurrent Java applications by utilizing exception handling mechanisms, user-defined exceptions, multithreading concepts, synchronization, and inter-thread communication.
CLO4	Utilize Java Collections Framework, file handling techniques, and JDBC to develop applications involving data management, file processing, and relational database connectivity.
CLO5	Design and implement event-driven graphical user interface (GUI) applications using AWT, Swing components, layout managers, and event handling mechanisms to develop interactive desktop applications.

Syllabus

Units	Content
1	**Introduction to Java and Object-Oriented Programming:** Introduction to Object-Oriented Programming (OOP), Features and Applications of Java, Java Program Structure, Java Virtual Machine (JVM), Java Runtime Environment (JRE), and Java Development Kit (JDK), Data Types, Variables, Constants, and Operators, Control Statements, Arrays in Java, Classes and Objects, Methods, Constructors, Method Overloading, Static Members, This Reference, Access (Visibility) Modifiers
2	**Object-Oriented Programming Concepts:** Inheritance: Types of Inheritance, the *super* Keyword, final Classes and Methods, Polymorphism: Method Overriding, Abstraction: Abstract Classes and Methods, Interfaces: Interfaces vs. Abstract Classes, Defining and Implementing Interfaces, Extending Interfaces, Packages: Creating Packages, Accessing Packages, Importing Packages

3	**Exception Handling and Multithreading:** Exception Handling: Benefits of Exception Handling, Exception Hierarchy, Checked and Unchecked Exceptions Exception Handling Mechanisms: try, catch, throw, throws, finally, User-defined Exceptions, Multithreading: Processes and Threads, Thread Life Cycle, Creating Threads, Thread Priorities, Thread Synchronization, Inter-thread Communication, Thread Interruption
4	**Collections, File Handling and JDBC:** Java Collections Framework, Introduction to Collections Framework, Collection Hierarchy, Common Collection Classes: ArrayList, Vector, Stack, Hashtable, Iterators, File Handling, Streams, Byte Streams, Character Streams, Text and Binary Input/Output, File Management using the *File* Class JDBC: Introduction to JDBC, JDBC Architecture, JDBC Drivers, Establishing Database Connectivity, Executing SQL Queries, Processing Result Sets, Updating Database Records using JDBC
5	**GUI Programming and Event Handling:** GUI Programming with Swing: Overview of AWT, Introduction to Swing, Swing vs. AWT, Swing Components: JButton, JLabel, JTextField, JTextArea, JCheckBox, JRadioButton, JMenu, Containers, JFrame, JPanel, JDialog, Layout Managers: Border Layout, Grid Layout, Event Handling: Introduction to Event Handling, Event Objects (Event Classes), Event Listeners, Handling Mouse Events, Handling Keyboard Events, Developing Event-Driven GUI Applications
S. No.	**Suggested Practicals / Mini Projects**
1	Write Java programs demonstrating the use of variables, data types, operators, control statements, arrays, and command-line arguments.
2	Develop Java programs using classes, objects, methods, constructors, static members, and this keyword.
3	Implement programs demonstrating method overloading using different parameter lists.
4	Write programs implementing different types of inheritance, the *super* keyword, and method overriding.
5	Develop Java programs using abstract classes, abstract methods, interfaces, multiple interface implementation, and interface inheritance.
6	Create user-defined packages, import packages, and demonstrate package accessibility in Java applications.
7	Write programs demonstrating exception handling using try, catch, throw, throws, and finally, including user-defined exceptions.
8	Develop multithreaded applications using the Thread class and the Runnable interface. Implement thread synchronization and demonstrate inter-thread communication.
9	Implement programs using Java Collections Framework, including ArrayList, Vector, Stack, and Hashtable.

10	Write programs to perform file operations using byte streams, character streams, text files, binary files, and the File class.
11	Develop Java applications to connect to a relational database using JDBC. Perform SQL operations including data insertion, retrieval, updating, and deletion using Statement and PreparedStatement.
12	Design GUI applications using Swing components such as JButton, JLabel, JTextField, JTextArea, JCheckBox, JRadioButton, JComboBox, JMenu, and JScrollPane.
13	Develop a mini-project integrating object-oriented programming concepts, exception handling, multithreading, collections, file handling, JDBC, and Swing-based GUI.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	POL4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	AVG CLO
CLO1	3	3	2	1	2	0	0	0	0	0	0	2	2.17
CLO2	3	3	3	2	2	0	0	0	0	0	0	3	2.67
CLO3	2	3	3	2	3	0	0	1	0	0	0	2	2.29
CLO4	2	3	3	2	3	0	0	0	0	0	0	3	2.57
CLO5	2	2	3	2	3	0	0	0	2	3	0	2	2.38
Avg PLO	2.40	2.80	2.80	1.80	2.60	0	0	1.00	2.00	3.00	0	2.40	2.39

Suggested Reading

1	*Head First Java — Kathy Sierra and Bert Bates, O'Reilly Media.*
2	*Object-Oriented Programming with Java — E. Balagurusamy, McGraw-Hill Education.*
3	*Thinking in Java — Bruce Eckel, Pearson.*
4	*Java: The Complete Reference — Herbert Schildt, McGraw-Hill Education.*
5	*Java: A Beginner's Guide — Herbert Schildt, McGraw-Hill Education.*

Teaching-Learning Strategies

Hands-on programming practice for implementing Java concepts, OOP, exception handling, collections, file handling, JDBC, and GUI applications.

Problem-based and demonstration-based learning through coding exercises, debugging, and application development.

Project-based learning through mini-project development integrating Java programming concepts.

Course Code	BCCSCSS326				3rd Semester	
Course Title	Signals and System				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)	
CLO1	Classify and analyze continuous-time and discrete-time signals and systems based on their fundamental properties.
CLO2	Determine the response of Linear Time Invariant (LTI) systems using convolution techniques and system representations.
CLO3	Apply Fourier Series, Fourier Transform, Laplace Transform and Z-Transform for the analysis of signals and systems. .
CLO4	Analyze sampled signals, signal reconstruction and basic digital filtering concepts for signal processing applications.
CLO5	Apply signals and systems concepts to solve engineering problems in Artificial Intelligence, Machine Learning, Image Processing, Computer Vision, Speech Processing and Cyber Security.

Syllabus	
Units	Content
1	**Introduction to Signals and Systems** Introduction to signals and systems, classification of continuous-time and discrete-time signals, deterministic and random signals, periodic and aperiodic signals, even and odd signals, energy and power signals, elementary signals (unit impulse, unit step, ramp, exponential and sinusoidal), signal operations including time shifting, time scaling and time reversal, classification of systems, properties of systems including linearity, time invariance, causality, memory, stability and invertibility
2	**Linear Time Invariant Systems and Convolution** Linear Time Invariant (LTI) systems, impulse response, convolution for continuous-time and discrete-time systems, graphical and analytical methods of convolution, properties of convolution, interconnection of systems, differential and difference equation models, system response and stability of LTI systems.

3	**Fourier Analysis** Fourier series representation of periodic signals, trigonometric and exponential Fourier series, Fourier transform of continuous-time signals, properties of Fourier transform, discrete-time Fourier transform (DTFT), frequency domain representation, convolution theorem, Parseval's theorem and applications of Fourier analysis in signal processing.
4	**Laplace and Z-Transforms** Laplace transform, region of convergence, properties of Laplace transform, inverse Laplace transform, transfer function, poles and zeros, stability analysis, Z-transform, region of convergence, properties of Z-transform, inverse Z-transform, relationship between Laplace and Z-transforms, applications in digital system analysis.
5	**Applications of Signals and Systems** Signal sampling theorem, aliasing, signal reconstruction, introduction to digital filters, correlation and autocorrelation, basics of random signals, introduction to spectral analysis, applications of signals and systems in Artificial Intelligence, Machine Learning, Image Processing, Speech Processing, Biomedical Signal Processing, Internet of Things and Cyber Security

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	3	2	1	1	0	0	0	0	0	0	1	1.83
CLO2	3	3	3	2	2	0	0	0	0	0	0	1	2.33
CLO3	3	3	3	2	2	0	0	0	0	0	0	2	2.50
CLO4	3	3	3	3	3	0	0	0	0	0	0	2	2.83
CLO5	2	3	3	2	3	0	0	0	0	0	0	3	2.67
Avg PLO	2.80	3.00	2.80	2.00	2.20	0	0	0	0	0	0	1.80	2.43

Suggested Reading

1	*Alan V. Oppenheim, Alan S. Willsky and S. Hamid Nawab, Signals and Systems, Pearson.*
2	*B. P. Lathi, Linear Systems and Signals, Oxford University Press.*
3	*Simon Haykin and Barry Van Veen, Signals and Systems, Wiley.*
4	*A. Anand Kumar, Signals and Systems, PHI Learning.*
5	*Hwei P. Hsu, Signals and Systems, Schaum's Outline Series, McGraw-Hill.*

Teaching-Learning Strategies

The course will be delivered through interactive classroom lectures, board work, multimedia presentations, numerical problem-solving sessions, tutorial discussions, simulation-based demonstrations using
MATLAB/Python, assignments, quizzes, case studies, application-oriented discussions and collaborative learning activities to strengthen analytical and problem-solving skills.

Course Code	BCCSLCP325				3rd Semester	
Course Title	Professional Practice and Digital Skills Laboratory				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	0	0	4	2	0	50

Course Learning Outcomes (CLOs)	
CLO1	Apply office productivity tools to prepare professional documents, spreadsheets, and presentations for technical and academic purposes.
CLO2	Utilize digital resources and professional communication practices to access, organize, and share technical information effectively.
CLO3	Apply basic version control techniques using Git and GitHub to manage project files and maintain technical documentation.
CLO4	Demonstrate professional practices by preparing technical reports, resumes, and project documentation while adhering to ethical, legal, and cybersecurity principles.
CLO5	Conduct a literature survey on a selected computing topic and effectively communicate technical concepts through a written report and seminar presentation.

Suggested lab exercises

Units	Content
1	**Office Productivity and Technical Documentation:** Prepare professional documents using a word processor by applying page formatting, tables, images, references, headers, footers, and templates. Create spreadsheets using formulas, sorting, filtering, and charts. Design and deliver professional presentations using presentation software.
2	**Internet Resources and Professional Communication:** Explore digital libraries, e-journals, online repositories, and e-learning platforms for academic resources. Create and manage professional emails, practice email etiquette, use cloud storage for file sharing and collaboration, and apply effective web search techniques for technical information retrieval.
3	**Version Control and Technical Documentation:** Install and configure Git, create and manage a local repository, perform basic version control operations (init, add, commit, status, and log), and publish a repository on GitHub. Prepare project documentation using Markdown, create a README file, and maintain version history for a mini project.

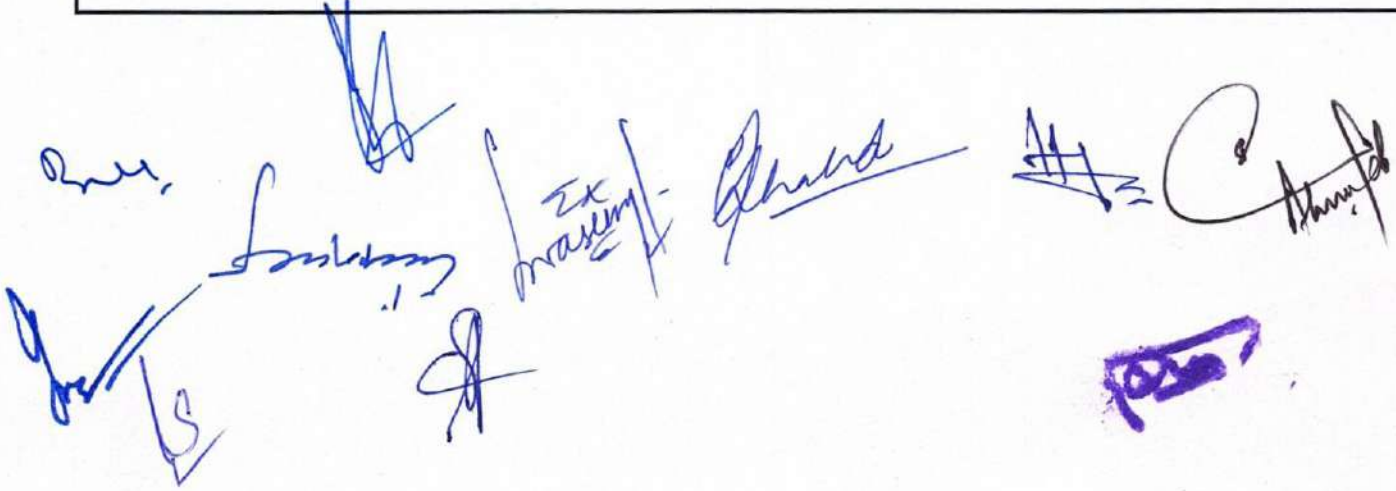
4	**Professional Practice and Ethics:** Prepare a professional resume/CV and technical report on a selected topic. Deliver a seminar presentation demonstrating effective communication skills. Practice teamwork, project planning, and time management while following professional ethics, academic integrity, cyber safety, data privacy, and responsible use of AI tools.
5	**Seminar Presentation:** Select a recent topic related to Computer Science, Artificial Intelligence, Data Science, or emerging technologies. Conduct a literature survey using authentic digital resources, prepare a technical report and presentation, deliver a seminar,

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	POL4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	1	1	2	0	3	0	0	0	1	3	0	2	1.86
CLO2	0	2	2	1	3	1	0	2	2	3	0	2	2.00
CLO3	1	2	2	1	3	0	0	1	2	2	1	2	1.70
CLO4	0	1	2	0	2	2	0	3	2	3	2	2	2.11
CLO5	2	2	2	2	2	0	0	1	2	3	1	3	2.00
Avg PLO	1.33	1.60	2.00	1.33	2.60	1.50	0	1.75	1.80	2.80	1.33	2.20	1.91

Teaching-Learning Strategies

Hands-on Laboratory Sessions with instructor demonstrations and guided practical exercises.
Project-Based Learning through the development of mini projects and technical documentation.
Problem-Based Learning involving practical computing tasks and real-world scenarios.
Seminar and Presentation on emerging computing technologies to enhance communication and presentation skills.
Collaborative and Peer Learning through team-based activities, discussions, and peer review.



4th Semester

Course Code	BCCSCPS426				4th Semester	
Course Title	Probability Statistics and Queuing				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)

CLO1	Calculate probabilities using Conditional probability, Rule of total probability and Bayes' theorem.
CLO2	Understand the concept of a random variable and the probability distributions
CLO3	To provide the required mathematical support in real-life problems and develop probabilistic models that can be used in several areas of science and engineering.
CLO4	Apply multivariate probability distributions, statistical distributions, regression, and correlation techniques to analyze and interpret data.
CLO5	Apply Markov chains, reliability concepts, and queuing models to solve stochastic and real-world engineering problems.

Syllabus

Units	Content
1	**Probability**: Random experiment, sample space, events, and classical statistical and axiomatic definitions of probability. Statements and proof of theorems on addition and multiplication of probabilities, simple problems. Bayes theorem on conditional probability
2	Measures of central tendency and measures of variations (dispersions), moments, measures of skewness, and kurtosis. **Random variables**: Derivation of the formula for mean variance and moments of random variables for discrete and continuous cases. probability mass, probability density and cumulative distribution functions, mathematical expectation, moments, probability and moment-generating functions, median and quantiles, Markov inequality, Chebyshev's inequality, problems.
3	**Multivariate Random Variables:** Joint, marginal and conditional probability distributions. Joint probability mass and density functions. Independence of random variables. Covariance and correlation. Conditional expectation and conditional variance. Joint moment-generating functions. Transformation of random variables. Bivariate normal distribution. Simple problems.

4	**Special Distributions**: Binomial, Poisson, and normal distributions; beta and gamma distributions. Sampling Distributions: t-distribution, f-distribution, chi-square distribution, and their applications. Methods of least squares, fitting a straight line and parabola of degree p. Regression and correlation. Multiple and partial correlations.
5	**Markov Chain and Reliability:** Markov chain, transition probabilities, limiting distributions, and concepts of reliability. Introduction to Queuing Theory: Markovian queuing models, Little's formula, application of queuing theory, and the M/M/1 Model.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	1	1	0	0	0	0	0	0	0	1	1.60
CLO2	3	2	2	2	0	0	0	0	0	0	0	1	2.00
CLO3	3	3	2	2	1	0	0	0	0	0	0	2	2.17
CLO4	2	3	2	3	2	0	0	0	0	1	0	2	2.14
CLO5	2	3	3	3	2	0	0	0	0	1	1	2	2.13
Avg PLO	2.60	2.60	2.00	2.20	1.67	0	0	0	0	1.00	1.00	1.60	2.01

Suggested Reading

1	Miller and Fread's Probability and Statistics for Engineers—Richard A. Johnson, Pearson Education Asia/PHI.
2	Fundamentals of Mathematical Statistics—S. C. Gupta and V. K. Kapoor, Sultan and Sons, Eastern Economy Edition.
3	Introduction to Mathematical Statistics—Craig and Hogg, Pearson.
4	Introduction to Probability and Statistics for Engineers and Scientists by Sheldon M. Ross.
5	Probability and Statistics with Reliability, Queuing with Reliability, and Queuing and Computer Science Applications by Kishor S. Trivedi, PHI.

Teaching-Learning Strategies

Interactive lectures supported by examples, visualizations, and problem-solving sessions.

Use of software tools for statistical analysis and simulation of probabilistic models.

Course Code	BCCSCDS426				4th Semester	
Course Title	Data Structures Using C				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	2	0	4	4	100	100

Course Learning Outcomes (CLOs)

CLO1	Understand the fundamentals of data structures, memory management, arrays, and strings.
CLO2	Implement recursion, stacks, and queues to solve computational problems
CLO3	Design and implement various linked lists and tree data structures along with their operations.
CLO4	Apply hash tables and graph data structures to solve the problems efficiently.
CLO5	To develop skills to apply appropriate data structures in problem-solving.

Syllabus

Units	
1	**Introduction:** Overview of data structures, classification of data structures: Primitive and non-primitive data types, abstract data types; Pointers and pointer arithmetic, Introduction to arrays, polynomial representation using arrays, limitations of arrays. String: introduction to strings, basic operations of strings.
2	**Recursion, Stack and Queue:** Introduction to recursion, recursive and iterative approaches; Stack: Static implementation, applications of Stack; Queue: Static implementation of queue, applications of queues, types of queues: linear, circular.
3	**Linked Lists and Trees:** Linked List: Linked list vs arrays, types of linked lists: Singly, doubly, circular, multiple linked lists; linked list operations: Insertion, deletion, searching and traversing, applications and limitations of linked lists, Dynamic Implementation of stack and queues. Tree: Binary tree, static and dynamic implementation, binary search tree; Introduction to AVL tree, B-Trees and B+ Trees. Heap: Max and Min heap, priority queue, heap sort.:

4	**Graphs and Searching Techniques:** Graph: Representation (adjacency matrix and adjacency list), breadth first search (BFS), depth first search (DFS), connected Components, introduction to topological sorting. Searching: Linear search and binary search, performance comparison.
5	**Sorting and Hashing:** Sorting Techniques: Bubble Sort, insertion sort, merge sort, quick sort, counting sort, Hashing: Hash functions, hash table, closed hashing (open addressing), linear Probing, quadratic probing, open hashing (separate chaining), applications of hashing.
S. No.	**Suggested Practicals / Mini Projects**
1	Arrays: Program to implement insertion and deletion operations in arrays
2	**Recursion**: Demonstrate the concept of recursions with the problem of the tower of Hanoi. Implement recursive sorting techniques- merge sort, quick sort.
3	**Stacks:** Static and dynamic implementation of push, pop and traversal operation in stack using both arrays and linked list. Implementation of stack to convert infix to postfix expression. Evaluating mathematical expressions using stack.
4	**Queues:** Static and dynamic implementation of linear queue and circular queue, operations: enqueue, dequeue and display.
5	**Linked List:** Implementation of insertion, search and deletion operations in singly linked list. Program to implement searching, insertion and deletion operations in doubly linked lists. Implementation of singly circular and doubly circular linked lists.
6	**Trees**: Program to create and traverse a binary tree recursively. Program to implement insertion and deletion operations in BST. Program to implement insertion and deletion operations in Heaps and Heap Sort. Program to implement insertion and deletion operations in AVL (with all rotations).
7	**Searching and sorting:** Program to implement different searching techniques - linear and binary search and compare their time complexities. Implement Binary search using recursion. Implementation of different sorting techniques like bubble, selection, insertion, merge, quick and heap sort.
8	**Graph**: Implement different representations of graphs (Adjacency list and adjacency matrix). Graph Traversal: Program to implement DFS/BFS of a graph.
9	**Hashing**: Program to implement linear probing, program to implement quadratic probing, program to implement double hashing, program to implement all the functions of a dictionary using hash

CLO-PLO Mapping Matrix													
CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO 10	PLO 11	PLO 12	Avg CLO
CLO1	3	2	1	0	1	0	0	0	0	0	0	1	1.6
CLO2	2	3	3	1	2	0	0	0	0	0	0	1	2.0
CLO3	2	3	3	2	2	0	0	0	0	0	0	1	2.2
CLO4	2	3	3	2	3	0	0	0	0	0	0	2	2.5
CLO5	2	3	3	2	3	0	0	0	0	0	0	3	2.7
Avg PLO	2.2	2.8	2.6	1.8	2.2	0	0	00	0	0	0	1.6	2.2

Suggested Reading

1	Fundamentals of Data Structures – Horowitz and Sahani, Galgotia Publication.
2	Reema Thareja, *Data Structures Using C*, Oxford University Press.
3	Narasimha Karumanchi (2016), Data Structures and Algorithms Made Easy: Data Structure and Algorithmic Puzzles, Second Edition, Career Monk publisher
4	An Introduction to Data Structures with applications – Jean Paul Trembley and Paul G. Sorenson, McGraw Hill Publications.
5	Data Structures – Lipschutz, Schaum's Outline Series, TMH.

Teaching-Learning Strategies

Interactive lectures supported by examples, visualizations/traces.
Problem-solving sessions involving the students to resolve problems using efficient data structure.
Tutorial assignments and classroom discussions.

Course Code	BCCSCCS426				4th Semester	
Course Title	Communication Systems				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	2	4	100	100

Course Learning Outcomes (CLOs)	
CLO1	Describe an electronic communication system along with the function of its various modules.
CLO2	Compare various analog modulation techniques.
CLO3	Appreciate the advantages of digital communication over analog and compare various digital modulation techniques.
CLO4	Understand the importance of quality, software configuration management, and compare various testing methods/techniques.
CLO5	Explain errors and its types and underscore the need for error correction in a digital communication system.

Syllabus	
Units	
1	**Analog Communication- AM:** Block diagram of a communications system (analog and digital); Channel; Types of channels: Guided and unguided; Baseband vs carrier communication; Need for modulation; Modulation and its types; Amplitude modulation (DSB-FC AM); Double sideband suppressed carrier (DSB-SC AM); Single side band (SSB AM); Vestigial side band modulation (VSB); Waveforms: Spectrum, bandwidth and power; Advantages/ disadvantages and applications of AM types.
2	**Analog Communication- FM:** Angle modulation; Frequency modulation (FM); Frequency deviation; Waveforms: Spectrum, bandwidth and power; Narrowband FM (NBFM) and wideband FM (WBFM); Advantages/ disadvantages and applications; Radio receivers; Super-heterodyne receiver: AM and FM; Intermediate frequency.

3	**Digital Communication:** Introduction to digital communication techniques; Advantages/ disadvantages with respect to analog communication; Concept of sampling; Sampling theorem and Nyquist rate; Types of sampling; Pulse analog modulation techniques: PAM, PWM, PPM; Intersymbol interference (ISI); Quantization and its types; Quantization error; Pulse Code Modulation (PCM); Bandwidth of PCM; Differential PCM (DPCM); Adaptive DPCM (ADPCM); Delta Modulation; Regenerative repeater; Generation, detection, waveforms and constellation diagrams. Pulse digital modulation techniques: ASK, FSK, PSK, BPSK, QPSK; QAM, 8 QAM, 16 QAM; OQPSK, M-ary schemes based on quadrature multiplexing; Frequency division multiplexing, Time division multiplexing. Examples and comparison of multiplexing schemes, Channel capacity and data rate
4	**Error analysis and Source coding Error:** Types, Bit error probability (BER), Hamming distance; Error detection (Parity check, checksum, CRC) and correction technique; Hamming code; Performance comparison.
5	**Introduction to noise and its types of noise: -** Shannon capacity theorem. Signal-to-noise ratio (SNR), SNR in AM, FM and PCM, Pre-emphasis and De-emphasis, Companding.

S. No.	Suggested Practicals
1	To study the generation of standard signals.
2	To study the spectrum Analysis of standard signals
3	To study the generation and detection of AM signals.
4	To study the generation of DSBSC amplitude modulated signals.
5	To study the detection of DSBSC amplitude modulated signals.
6	To study the generation of frequency modulated signals.
7	To study the detection of frequency modulated signals.
8	To measure sensitivity, selectivity, and fidelity of a radio receiver.
9	To measure the noise figure of AM and FM System.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO 9	PLO 10	PLO 11	PLO 12	Avg CLO
CLO1	3	2	1	1	2	0	0	0	0	1	0	1	1.57
CLO2	3	3	2	2	2	0	0	0	0	1	0	1	2.00

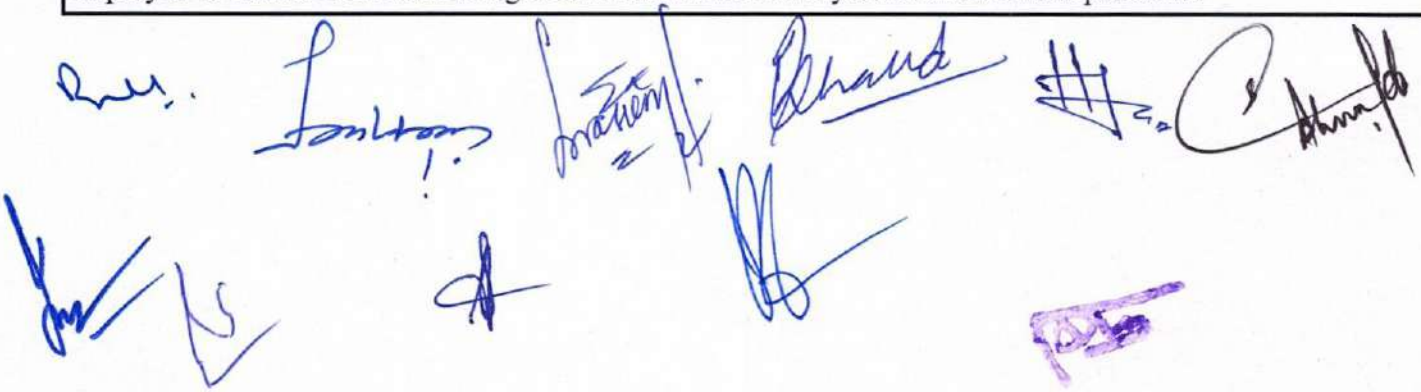
CLO3	3	3	2	3	2	0	0	0	0	1	0	1	2.14
CLO4	3	3	3	2	3	1	0	0	1	2	0	2	2.22
CLO5	3	3	3	2	3	2	0	2	1	2	1	2	2.18
Avg PLO	3.0	2.8	2.2	2.0	2.4	1.5	0	2.0	1.0	1.4	1.0	1.4	2.02

Suggested Reading

1	B. P. Lathi and Z. Ding, Modern Digital and Analog Communication Systems, 4th Edition, Oxford University Press, 2011.
2	Taub & Schilling: Principles of communication systems- McGraw-Hill Education (India).
3	S. Haykin, Communication Systems, 4th Edition, Wiley, 2006
4	G. Kennedy, Electronic Communication Systems

Teaching-Learning Strategies

The course will be delivered through interactive classroom lectures, multimedia presentations for signal analysis and network architectures, tutorial sessions for problem-solving, and hands-on laboratory experiments. Key strategies include simulation of modulation and line coding schemes, network design and configuration using simulators like Cisco Packet Tracer, implementation of error detection techniques, and analysis of transmission media. Collaborative learning, assignments, quizzes, and mini-projects will be employed to reinforce understanding of data communication systems and network protocols.



Course Code	BCCSCAD426				4th Semester	
Course Title	Analysis and Design of Algorithms				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	2	4	100	100

Course Learning Outcomes (CLOs)

CLO1	Analyze algorithm efficiency using asymptotic notations and recurrence relations.
CLO2	Design algorithms using greedy, divide-and-conquer, and dynamic programming techniques.
CLO3	Apply backtracking and branch-and-bound methods to solve optimization problems.
CLO4	Implement graph algorithms for traversal, shortest paths, and connectivity problems.
CLO5	Evaluate computational problems using complexity theory and appropriate algorithmic paradigms.

Syllabus

Units	Content
1	**Introduction**: Algorithm design paradigm, concept of algorithmic efficiency, run time analysis of Algorithms, asymptotic notations, and recurrence relations, master's theorem. Randomized Algorithms: Identifying the repeated element, Primality testing, Advantages and Disadvantages.
2	**Greedy Method:** Overview of the greedy paradigm, minimum cost spanning tree, Prim's algorithm,Kruskal's algorithm, approximate solution (knapsack problem), and single source shortest path, dijkstra's algorithm. **Divide and Conquer:** Structure of divide and conquer algorithms, binary search and its analysis, analysis of quick sort.
3	**Dynamic Programming:** Overview, Divide and Conquer vs Dynamic programming, 0/1 knapsack, FloydWarshall's algorithm, matrix multiplication, travelling salesman problem, longest common subsequence, Bellman ford algorithm.
4	**Backtracking**: Overview, N-queen problem, knapsack problem, graph colouring, Hamiltonian cycles.**Branch and Bound:** LC searching, bounding, FIFO branch and bound, Job scheduling, 0/1 knapsackproblem, travelling salesperson problem.Comparative analysis of Divide and conquer and dynamic programming, backtracking and dynamic programming

5	**Application of Graph Traversal Techniques:** Topological sorting of DAGs, biconnected components,and strongly connected components in directed graphs.**Computational Complexity:** Complexity measures, polynomial time complexity, non-polynomial time complexity, NP hard and NP complete classes, polynomial time reduction, Cook's Theorem.
S. No.	**Suggested Practicals / Mini Projects**
1	Write programs to compute the running time of iterative and recursive algorithms. Compare empirical and theoretical time complexities.
2	Implement recursive algorithms and verify time complexity using recurrence relations. Implement randomized algorithms for repeated element detection and primality testing. Compare deterministic and randomized approaches
3	**Greedy method:**Implement Prim's Algorithm. Implement Kruskal's Algorithm.Implement Dijkstra's Single Source Shortest Path Algorithm. Implement Fractional Knapsack Algorithm.
4	**Divide and Conquer:** Implement Binary Search. Implement Quick Sort. Analyze their time complexity for best, average, and worst cases.
5	**Dynamic Programming:** Implement 0/1 Knapsack. Implement Matrix Chain Multiplication. Implement Longest Common Subsequence (LCS). Implement Floyd Warshall Algorithm. Implement Bellman Ford Algorithm. Implement Travelling Salesperson Problem. Implement Fibonacci series using recursion and dynamic programming and compare the time complexity
6	**Backtracking:** Implement N-Queen Problem. Implement Graph Coloring. Implement Hamiltonian Cycle.
7	**Branch and Bound:** Implement LC Branch and Bound. Implement FIFO Branch and Bound. Implement Job Scheduling. Implement 0/1 Knapsack using Branch and Bound. Implement Travelling Salesperson Problem using Branch and Bound.
8	**Graph Algorithms** Implement Topological Sorting. Implement Strongly Connected Components. Implement Biconnected Components.

CLO-PLO Mapping Matrix

CLO/PLO	PLO 1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO 9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	3	2	3	2	0	0	0	0	1	0	1	2.14
CLO2	3	3	3	2	3	0	0	0	0	1	0	1	2.29

CLO3	3	3	3	3	3	0	0	0	0	1	0	1	2.43
CLO4	3	3	3	3	3	0	0	0	1	2	0	2	2.43
CLO5	3	3	3	3	2	1	0	0	0	2	0	2	2.38
Avg PLO	3.0	3.0	2.8	2.8	2.6	1.0	0	0	1.0	1.4	0	1.4	2.33

Suggested Reading	
1	Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill
2	Fundamentals of Algorithms E. Horowitz et al.
3	Algorithm Design, 1st Edition, Jon Kleinberg and Éva Tardos, Pearson.
4	Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.
5	The Design and Analysis of Computer Algorithms,Aho, Hopcroft and Ullman, Pearson.

Teaching-Learning Strategies
Interactive lectures supported by examples, visualizations, and derivations of formal computational models.
Problem-solving sessions involving the design and analysis of algorithms using greedy, divide-and-conquer, dynamic programming, backtracking, and branch-and-bound techniques.
Collaborative project-based learning and tutorial assignments regarding algorithm efficiency, complexity theory, and graph traversal.
Hands-on laboratory sessions implementing and analyzing various algorithms using programming tools.

Course Code	BCCSCMP426				4th Semester	
Course Title	Microprocessor				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	NIL

Course Learning Outcomes (CLOs)

CLO1	Explain the internal architecture, pin configuration, and bus organization of 8085/8086 microprocessors.
CLO2	Develop and debug assembly language programs using the 8085/8086 instruction set.
CLO3	Design memory and I/O interfacing circuits for microprocessor-based systems.
CLO4	Analyze interrupt handling mechanisms, DMA operations, and serial/parallel communication.
CLO5	Compare advanced microprocessors/microcontrollers and apply microprocessor concepts to real-world embedded applications.

Syllabus

Units	
1	**Introduction to Microprocessors and 8085 Architecture** Evolution of microprocessors: basic microcomputer system and block diagram; 8085 microprocessors: architecture, register organization, ALU, flag register; Pin diagram and signal description of 8085; Bus organization; Memory and I/O interfacing concept (overview), memory map; Instruction cycle, machine cycle, and timing diagram of 8085; Addressing modes of 8085 microprocessor.
2	**8085 Instruction Set and Assembly Language Programming** Instruction set classification: data transfer, arithmetic, logical, branching, machine control instructions; Assembly language programming; Programs using arithmetic and logical instructions; Programs involving loops, branching, and call-return operations; Stack and subroutines: PUSH, POP, CALL, RET instructions; Sorting and searching programs using 8085 assembly language.
3	**Memory and I/O Interfacing** Semiconductor memory organization: RAM, ROM, EPROM interfacing; Memory interfacing techniques and address decoding logic; I/O interfacing: memory-mapped I/O and I/O-mapped I/O; Programmable Peripheral Interface 8255, Interfacing 8255 with 8085 for input/output applications; Programmable Interval Timer 8253/8254; Case study: stepper motor and traffic light controller interfacing.

4	**Interrupts, DMA, and Serial Communication** Interrupt structure of 8085: hardware and software interrupts, Vectored and non-vectored interrupts, priority of interrupts; Programmable Interrupt Controller 8259; Direct Memory Access (DMA), DMA controller 8257; Serial communication concepts: synchronous vs asynchronous, Programmable Communication Interface 8251 (USART), Interfacing 8251 with 8085 for serial data transfer, Comparison of polling, interrupt-driven, and DMA-based data transfer.
5	**16-bit Microprocessors and Advanced Processors** 8086 microprocessors: architecture, BIU and EU functional units, register organization, flag register, and memory segmentation in 8086, Pin diagram and signal description of 8086, minimum and maximum mode, Addressing modes and instruction set of 8086 (overview); Interrupts in 8086; Introduction to 80286/80386 architecture (overview); Overview of microcontrollers (8051) and comparison with microprocessors; Applications of microprocessors in embedded systems and emerging trends.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	POL4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	AVG CLO
CLO1	3	2	0	0	1	0	0	0	0	0	0	0	2
CLO2	2	3	2	0	2	0	0	0	0	0	0	0	2.25
CLO3	2	2	3	2	3	0	0	0	0	1	0	0	2.17
CLO4	2	2	2	3	3	1	0	1	2	2	0	0	2.13
CLO5	1	2	2	2	3	1	0	2	2	3	1	2	1.82
Avg PLO	2	2.2	2.25	2.33	2.4	1	0	1.5	2	2	1	2	1.76

Suggested Reading

1	Ramesh S. Gaonkar, "Microprocessor Architecture, Programming, and Applications with the 8085", Penram International Publishing.
2	Douglas V. Hall, "Microprocessors and Interfacing: Programming and Hardware", McGraw Hill Education.
3	Barry B. Brey, "The Intel Microprocessors: Architecture, Programming and Interfacing", Pearson Education.
4	Yu-Cheng Liu and Glenn A. Gibson, "Microcomputer Systems: The 8086/8088 Family", PHI Learning.
5	E-resources: NPTEL Course: "Microprocessors and Microcontrollers" – IIT Madras (nptel.ac.in)

Teaching-Learning Strategies

The course is delivered through interactive lectures, demonstrations, laboratory sessions, and programming exercises. Students gain practical experience using microprocessor simulators and hardware kits to develop assembly language programs and interface memory and I/O devices. Assignments and mini-projects reinforce theoretical concepts through real-world applications.

Course Code	BCCSLPP426				4th Semester	
Course Title	Python Programming				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	0	0	4	2	0	50

Course Learning Outcomes (CLOs)	
CLO1	Develop Python programs using fundamental programming constructs, including input/output operations, variables, data types, operators, expressions, conditional statements, loops, functions, recursion, and file handling with appropriate exception handling.
CLO2	Implement Python programs utilizing built-in data structures (strings, lists, tuples, sets, and dictionaries), object-oriented programming concepts (classes, objects, inheritance, polymorphism, and encapsulation), and object serialization techniques to solve computational problems.
CLO3	Apply Python scientific computing libraries such as NumPy and Pandas to perform array manipulation, data preprocessing, statistical analysis, data transformation, and efficient handling of structured datasets.
CLO4	Create meaningful data visualizations using Matplotlib and Seaborn, and develop introductory machine learning and image processing applications using libraries such as Scikit-learn, TensorFlow, PyTorch, and OpenCV.
CLO5	Design, develop, and evaluate a real-world Python based application by integrating programming fundamentals, data analysis, visualization, and introductory machine learning techniques to address practical problems

Suggested Practicals / Mini Projects	
Units	Content
1	Write Python programs to demonstrate basic input/output operations, variables, data types, operators, expressions, and assignment statements.
2	Write Python programs using conditional statements (if, if-else, nested if-else), looping statements (for, while, nested loops), and loop control statements (break, continue, pass).
3	Write Python programs to Implement Python functions using positional, keyword, and default arguments, multiple return values, variable scope, and recursion.
4	Write Python programs to perform operations on strings, lists, tuples, sets, and dictionaries using built-in functions and methods.
5	Write Python programs to Implement file handling operations (text files), exception handling (try-except-finally), and object serialization using pickling.

6	Write Python programs using NumPy for array creation, indexing, slicing, broadcasting, mathematical operations, searching, sorting, and statistical functions.
7	Write Python programs to perform data analysis using Pandas by creating Series and DataFrames, reading and writing CSV files, handling missing data, sorting, filtering, and generating descriptive statistics.
8	Write Python programs to implement object-oriented programming concepts including classes, objects, constructors, inheritance, method overriding, data hiding (encapsulation), and function overloading.
9	Develop applications using Python libraries such as Scikit-learn, TensorFlow, PyTorch, and OpenCV for basic machine learning and image processing tasks.
10	Mini Project: Develop a real-world application integrating Python programming, NumPy, Pandas, Object-Oriented Programming, data visualization, and introductory machine learning concepts.
11	Create data visualizations using Matplotlib and Seaborn, including line plots, bar charts, pie charts, histograms, scatter plots, subplots, heatmaps, and distribution plots.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	POL4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	AVG CLO
CLO1	3	2	2	1	3	0	0	0	0	0	0	1	2.00
CLO2	3	3	3	2	3	0	0	0	1	0	0	2	2.43
CLO3	2	3	2	3	3	0	0	0	0	1	0	2	2.29
CLO4	2	2	3	2	3	0	0	0	1	2	0	2	2.14
CLO5	2	3	3	3	3	1	0	1	2	2	1	3	2.20
Avg PLO	2.40	2.60	2.60	2.20	3.00	1.00	0	1.00	1.33	1.67	1.00	2.00	2.21

Suggested Reading

1	Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
2	Learning Python, Mark Lutz, Orielly.
3	Think Python, Allen Downey, Green Tea Press.
4	Core Python Programming, W.Chun, Pearson.

5	Introduction to Python, Kenneth A. Lambert, Cengage.

Teaching-Learning Strategies
Seminars and Presentations on Python applications, data science, and emerging technologies.
Problem-Based Learning involving coding challenges and real-world programming tasks.
Project-Based Learning through Python mini projects and application development.