5th Semester

Course Code	BCCSCOS526				5th Semester	
Course Title	Operating Systems				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	2	1	2	4	100	100

Course Learning Outcomes (CLOs)	
CLO1	Introduces operating system concepts which include process, scheduling, synchronization deadlocks, memory management and I/O management.
CLO2	Provides insight of issues to be considered in the design and development of operating systems.
CLO3	Introduces Unix commands, system call interface for process management, communication and I/O in Unix.
CLO4	Understand the fundamentals of Disk management and I/O devices.
CLO5	Simulate operating system concepts like scheduling, deadlock, etc.

Syllabus

Units	Content
1	**Introduction to operating systems:** Operating system functions, evolution of operating systems, batch, interactive, time sharing and real-time system, system protection. Operating system structure: System components, operating system services. Process management: Process concept, principle of concurrency, producer consumer problem, critical section problem, semaphores, inter-process communication.
2	**CPU scheduling:** Scheduling concept, scheduling criteria, scheduling algorithms, multiprocessor scheduling. **Deadlock:** System model, deadlock characterization, prevention, avoidance, detection and recovery.
3	**Memory management:** Base machine, Resident monitor, Multiprogramming with fixed partition, Multiprogramming with variable partition, Multiple base register, Paging, Segmentation, Virtual memory concept, Demand paging, Performance, Paged replaced algorithm, Allocation of frames, Thrashing; Cache memory, Organization, Impact on performance
4	**Virtual memory management:** Virtual memory concept, Demand paging, Performance, Page replacement algorithms, Allocation of frames, Thrashing. **File system:** File Concept, File Organization and Access Mechanism, File Directories and File Sharing.

5	**Disk management:** Disk Structure, Disk I/O, Disk scheduling algorithms, RAID Structure. **I/O management:** I/O devices and organization of I/O function, I/O Buffering, DISK I/O, Operating System Design Issues.
S. No.	**Suggested Practicals / Mini Projects**
1	**Execution of basic UNIX commands:** • File/Directory Handling commands. • Explore system variables such as PATH, HOME. • Modifying File Access Permissions and identifying different types of users in UNIX.
2	**Filters and I/O Redirection:** • Demonstrate all features of awk, sed and grep commands. Compare the results of these commands. • Execute commands related to inode, I/O redirection, piping.
3	**Shell Programming:** shell script exercise based on following: • Interactive shell script, Positional parameters, Arithmetic and Logical operators, If structure • While, for, until loop, Meta characters.
4	**Inter-process Communication:** • Write a program to demonstrate a one-way pipe between two Processes. • Write a program to illustrate IPC through pipe and fork system calls –Printing only odd numbers.
5	**Simulation of scheduling algorithms:** Write a shell program to implement the following process scheduling algorithms • First Come First Serve • Shortest Remaining Job First • Round Robin
6	**Implement the following:** The Producer – Consumer problem using semaphores (using UNIX system calls). Banker's algorithm for deadlock avoidance. Paging and Segmentation.
7	**Mini Project/Lab Assignment.**
	Note: This is only a suggested list of lab exercises. Instructors may frame additional exercise relevant to the course contents.

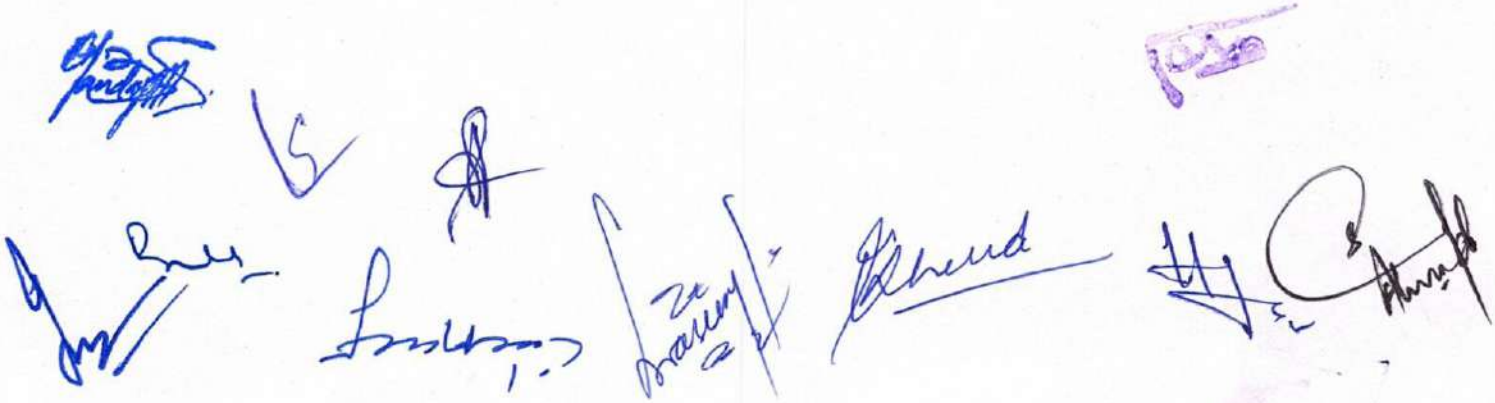
CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	0	0	0	0	0	0	0	0	0	1	2
CLO2	3	3	3	2	1	0	0	0	0	0	0	1	2.17
CLO3	3	2	2	1	3	0	0	0	0	0	0	1	2
CLO4	3	2	1	2	2	0	1	0	0	0	0	1	1.71

CLO5	**3**	**3**	**3**	**3**	**3**	**0**	**0**	**0**	**0**	**0**	**0**	**2**	**2.43**
Avg PLO	**3.0**	**2.4**	**2.3**	**2.0**	**2.3**	**0.0**	**1.0**	**0**	**0**	**0**	**0**	**1.2**	**2.1**

Suggested Reading	
1	*Silberschatz, A., Peterson, J. L., & Galvin, P. B. (1991). Operating system concepts. Addison-Wesley Longman Publishing Co., Inc...*
2	*Andrew, S. T., & Herbert, B. (2015). Modern operating systems. Pearson Education.*
3	*Stallings, W. (1998). Operating systems internals and design principles. Prentice-Hall, Inc...*
4	Bhatt, P. C. P. (2019). An Introduction to Operating Systems: Concepts and Practice (GNU/Linux and Windows). PHI Learning Pvt. Ltd...
5	*Deitel, H. M., Deitel, P. J., & Choffnes, D. R. (2004). Operating systems. Pearson/Prentice Hall.*

Teaching-Learning Strategies
Lectures and interactive classroom discussions.
Demonstrations and hands-on experiments on Open-Source Operating Systems.
Shell programming to understand the searching and filtering operations.



Course Code	BCCSCCN526				Semester: 5th	
Course Title	Computer Networks				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	2	4	100	100

Course Learning Outcomes (CLOs)

CLO1	Understand the fundamental concepts of computer networks, including network architectures, transmission techniques, network topologies.
CLO2	Analyze data link layer functions including flow control, error control and medium access control techniques.
CLO3	Apply network layer concepts to configure and manage logical addressing, subnetting, packet handling, and routing algorithms for efficient data delivery.
CLO4	Explain the functions and protocols of the transport layer, including TCP, UDP, SCTP, connection management, flow control, and reliable end-to-end communication services.
CLO5	Evaluate network performance and application-layer services by studying congestion control mechanisms, Quality of Service (QoS) techniques, and key Internet applications such as DNS, HTTP, email, file transfer, and remote access services.

Syllabus

Units	Content
1	**Basic Concepts of Networks:** Introduction, history and development of computer networks, Data transmission techniques: simplex, half duplex, full duplex; Types of networks (LAN, MAN and WAN); Different network topologies like bus, star, ring, hybrid. **Network Protocol Architecture:** OSI Reference model, Layers of OSI model: Physical, Data-link, Network, Transport, Session, Presentation and Application layer; Concept of TCP/IP Protocol Suite. **Physical Layer:** Different types of transmission media; role of repeaters; Line Encoding Schemes (NRZ, NRZI, Manchester, 4B/5B, etc.)

2	**Flow and Error Control:** Stop and wait flow control, sliding window flow control, error control protocols, ARQ techniques, Stop-and-wait ARQ, Goback-N ARQ, Selective repeat ARQ. HDLC Protocol, Point to Point Protocol, Ethernet, Token Ring network, WLANs **Medium Access Control Protocols:** Random Access: ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA; Controlled Access: Polling, token passing, reservation. **Network Switching Techniques:** Circuit, message and packet switched networks.
3	**Network layer:** Logical Addressing-IPv4 and IPv6 addresses, Concept of Sub-netting and Classless Addressing, Network Address Translation. **IP Packets:** IPv4 and IPv6 packet format, Packet Fragmentation **Routing Algorithms:** Routing tables, features of a routing algorithm, classification; concept of optimality principle, sink tree, flooding, fixed routing, random routing, adaptive routing; shortest path algorithm, Dijkstra algorithm, distance vector and link state algorithm.
4	**Transport Layer:** Services provided to the upper layers, addressing, Socket Addresses, connection establishment, Connection release, Error Control & Flow Control **Protocols:** User Datagram Protocol (UDP), Introduction to TCP, The TCP Service Model, The TCP Segment Header, The Connection Establishment, The TCP Connection Release, The TCP Sliding Window **Introduction to SCTP:** SCTP Service, SCTP Features, Packet Format, SCTP Association
5	**Congestion Control:** Congestion in networks, congestion control strategies, Congestion control in TCP and Frame Relay; **Quality of Service (QoS):** Techniques to improve QoS **Application Layer:** Client server model, File Transfer, HTTP, Electronic mail, Remote Logging.

S. No.	Suggested Practicals / Mini Projects
1	Introduction of LANs and Network Wire Crimping.
2	Experimental study of assigning IP addresses to nodes and sharing resources over the networks.
3	Experiments with packet sniffers to study the TCP/IP protocols.
4	Introduction of Network Simulators.
5	Introduction to NS2 (network simulator) - small simulation exercises to study behaviour of various protocols.
6	Setting up a small IP network: Configure interfaces, IP addresses and routing protocols.
7	Queuing disciplines.
8	Small exercises in socket programming in high level languages.

9	Note: This is only a suggested list of lab exercises. Instructors may frame additional exercise relevant to the course contents.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	0	0	0	0	0	0	0	0	0	1	2
CLO2	3	3	2	2	1	0	0	0	0	0	0	1	2
CLO3	3	3	3	2	2	0	0	0	0	0	0	1	2.33
CLO4	3	2	2	1	1	0	0	0	0	0	0	1	1.67
CLO5	3	3	2	3	2	1	1	0	0	0	0	2	2.13
Avg PLO	3.0	2.6	2.3	2.0	1.5	1.0	1.0	0	0	0	0	1.2	2.0

Suggested Reading

1	*William Stallings: Data and Computer Communications, PHI*
2	*D.P Bertsekas "Data Networks" Prentice Hall*
3	*Andrew Tanenbaum, "Computer Networks", Pearson*
4	*J. F. Kurose "Computer Networking" Addison-Wesley*
5	*Douglas E. Comer, "Internetworking with TCP/IP, Volume1", PHI*
6	*Peterson and Davie, "Computer Networks", Morgan Kaufman*

Teaching-Learning Strategies

Lectures and interactive classroom discussions.

Demonstrations and hands-on laboratory experiments using Network Simulators.

Project-based learning through the Implementation and evaluation of Networking Protocols.

Course Code	BCCSCTC526				Semester: 5th	
Course Title	Theory of Computation				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 concepts of formal languages, finite automata, regular expressions, and regular grammars.
CLO2	Analyze context-free grammars, pushdown automata, and their applications in language recognition.
CLO3	Apply Turing machine models to solve computational problems and classify formal languages.
CLO4	Analyze decidability, reducibility, and undecidability problems using formal computational models.
CLO5	Evaluate computational complexity, NP-completeness, and related problems using complexity theory principles.

Syllabus	
Units	Content
1	**Introduction to Theory of Computation, Regular Languages and Finite Automata:** Introduction to Theory of Computation: Significance and applications of Theory of Computation in Computer Science; Mathematical Models of Computation; Formal Languages, Alphabets, Strings and Languages; Operations on Languages; Deterministic Finite Automata (DFA); Non-Deterministic Finite Automata (NFA); Equivalence of DFA and NFA; Finite Automata with ε-transitions; Regular Languages; Regular Expressions; Conversion between Finite Automata and Regular Expressions; Closure Properties of Regular Languages; Pumping Lemma for Regular Languages; Applications of Finite Automata.
2	**Context-Free Languages and Pushdown Automata:** Context-Free Grammars (CFGs); Derivations and Parse Trees; Ambiguous and Unambiguous Grammars; Simplification of CFGs; Chomsky Normal Form (CNF); Greibach Normal Form (GNF); Pushdown Automata (PDA); Deterministic and Non-Deterministic Pushdown Automata (DPDA and NPDA); Acceptance by Empty Stack and Final State; Equivalence of CFGs and PDAs; Closure Properties of Context-Free Languages; Pumping Lemma for Context-Free Languages

3	**Turing Machines and Computability:** Turing Machines; Instantaneous Descriptions; Design of Turing Machines; Variants of Turing Machines; Deterministic and Non-Deterministic Turing Machines (DTM and NDTM); Universal Turing Machine; Church-Turing Thesis; Recursive and Recursively Enumerable Languages; Introduction to Computability.
4	**Undecidability:** Decidable and Undecidable Problems; The Halting Problem; Post Correspondence Problem (PCP); Mapping Reducibility; Undecidability Proof Techniques; Examples of Undecidable Problems involving Turing Machines and Context-Free Languages.
5	**Computational Complexity:** Time Complexity and Space Complexity; Complexity Classes P and NP; Polynomial-Time Verification; Polynomial-Time Reductions; NP-Hard and NP-Complete Problems; Cook's Theorem; Satisfiability (SAT), Clique, Vertex Cover and Traveling Salesman Problems. Introduction to Quantum Computing and Quantum Automata.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	0	0	0	0	0	0	0	0	0	1	2.0
CLO2	3	3	2	2	0	0	0	0	0	0	0	1	2.2
CLO3	3	3	3	2	1	0	0	0	0	0	0	1	2.17
CLO4	3	3	2	3	1	0	0	0	0	0	0	1	2.17
CLO5	3	3	2	2	2	0	0	0	0	0	0	2	2.33
Avg PLO	3.0	2.8	2.3	2.25	1.33	0	0	0	0	0	0	1.2	2.17

Suggested Reading

1	*John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education.*
2	*Michael Sipser, Introduction to the Theory of Computation, Cengage Learning.*
3	*Peter Linz, An Introduction to Formal Languages and Automata, Jones & Bartlett Learning.*
4	*John C. Martin, Introduction to Languages and the Theory of Computation, McGraw-Hill Education.*

5	*K. L. P. Mishra and N. Chandrasekaran, Theory of Computer Science, PHI Learning.*

Teaching-Learning Strategies
Interactive lectures supported by examples, visualizations, and derivations of formal computational models.
Problem-solving sessions involving the design and analysis of finite automata, grammars, pushdown automata, and Turing machines.
Tutorial assignments and classroom discussions on language properties, computability, undecidability, and computational complexity.
Case studies and seminars highlighting applications of Theory of Computation in compiler design, formal verification, artificial intelligence, and emerging computational paradigms.

Course Code	BCCSCCA526				Semester: 5th	
Course Title	Computer Organization and Architecture				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)

CLO1	Explain the basic structure of a computer system consisting of CPU, main memory, I/O and their functional interconnection.
CLO2	Understand and apply register transfer logic to describe the microoperations in the execution of an instruction.
CLO3	Analyze instruction format to appreciate its implications on opcode and addressing modes.
CLO4	Describe how integers and floating-point numbers are represented and illustrate how arithmetic is performed.
CLO5	Appreciate the hierarchical memory organization in a computer system and compare various cache mapping techniques.

Syllabus

Units	Content
1	**Basic structure and functional interconnection:** Basic structure and functions of a digital computer, Basic functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Von Neuman architecture, hierarchical structure of functional units and their interconnection; System bus. Structure of CPU. CPU registers and their functions viz. Program Counter, Instruction Register, Memory Address Register, Memory Buffer Register, Program Status Word, etc. Execution of a complete (basic) instruction. Instruction set (ISA), Instruction cycle.
2	**Instruction format and addressing modes:** Various addressing modes, Visualization of addressing modes in high level program constructs viz. array indexing. Instruction format and types, Opcode and operands. Micro-operations and Register Transfer Language, CPU organizations- accumulator, general register and stack, Implications on zero, one, two and three address instructions. CISC and RISC architecture.

3	**Data representation and arithmetic:** Signed number representation, fixed and floating-point representations, character representation. Computer arithmetic - integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication - shift-and-add, Booth multiplier, carry save multiplier, etc. Division - non-restoring and restoring techniques, floating point arithmetic. Control Unit Organization: Hardwired control, micro-programmed control, micro-instructions, address sequencing, control memory.
4	**Memory hierarchy and cache memory organization:** Semiconductor main memory, RAM and its types-SRAM and DRAM, memory hierarchy, locality of reference, cache memory, organization and characteristics, mapping techniques- direct, associative and set associative, replacement algorithms (brief account only), write policies- write-back and write-through.
5	**I/O organization and performance metrics:** I/O modules and their functions, programmed I/O, commands, memory-mapped and isolated I/O, interrupt-driven I/O, DMA, Introduction to standard interfaces viz. PCI, SCSI, and USB. CPU equation and Amdahl's law. Introduction to pipelining.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	1	0	0	0	0	0	0	0	0	0	1	1.67
CLO2	3	2	1	0	0	0	0	0	0	0	0	0	2.00
CLO3	2	3	0	0	0	0	0	0	0	0	0	0	2.50
CLO4	3	2	0	0	0	0	0	0	0	0	0	0	2.50
CLO5	3	2	2	0	0	0	0	0	0	0	0	1	2.00
Avg PLO	2.80	2.00	1.5	0	0	0	0	0	0	0	0	1.00	2.13

Suggested Reading

1	*Computer Organisation and Architecture, William Stallings, Pearson Education.*
2	*David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Elsevier.*
3	*Computer System Architecture, J. P. Hayes, Pearson Education*
4	*Computer System Architecture, M. Moris Mano, Pearson Education.*

5	*The Essentials of Computer Organisation and Architecture, Null and Lobur, Jones and Bartlett Publishers, Inc*

Teaching-Learning Strategies
Lectures and interactive discussions.
Illustrative presentations for visualizing internal computer processes.
Student seminars on contemporary processor architectures.

Course Code	BCCSCAM526				Semester: 5th	
Course Title	AI and Machine Learning				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 fundamentals, history, applications, and agent-based approach of Artificial Intelligence.
CLO2	Apply uninformed, heuristic, and adversarial search techniques for solving AI-based problems.
CLO3	Analyze knowledge representation, logical reasoning, and probabilistic reasoning techniques.
CLO4	Apply supervised and unsupervised Machine Learning algorithms for classification and clustering problems.
CLO5	Explain neural networks, genetic algorithms, fuzzy logic, and basic NLP techniques used in AI and ML applications.

Syllabus

Units	Content
1	**Introduction to Artificial Intelligence and Uninformed Search:** Introduction, history and applications of Artificial Intelligence. Intelligent agents, structure of intelligent agents and problem-solving agents. Uninformed search strategies: Breadth First Search, Depth First Search, Depth Limited Search, Uniform Cost Search and Iterative Deepening Search.
2	**Heuristic and Adversarial Search Techniques:** Heuristic search, heuristic functions, hill climbing, beam search, A* algorithm, completeness and optimality of search algorithms. Adversarial search, game playing, minimax algorithm, alpha-beta pruning and evaluation functions.
3	**Knowledge Representation and Reasoning:** Propositional logic, first-order logic, inference in first-order logic, forward chaining, backward chaining, theorem proving, resolution theorem proving, rules and rule-based reasoning, structured knowledge representation. Reasoning under uncertainty, probabilistic reasoning, Bayesian inference networks and Hidden Markov Models.
4	**Machine Learning Fundamentals and Algorithms:** Introduction to Machine Learning, supervised learning, unsupervised learning and reinforcement learning. Data preprocessing, training and testing, model evaluation. Classification techniques: Support Vector Machines, Decision Trees and Naive Bayes.

	Clustering techniques: K-Means clustering, Gaussian Mixture Model and Expectation-Maximization algorithm.
5	**Neural Networks, Soft Computing and Natural Language Processing:** Introduction to neural networks, perceptron, activation functions and basic neural network architectures. Genetic algorithms: operations, applications and limitations. Fuzzy logic: fuzzification, fuzzy sets, operations on fuzzy sets, hedges, reasoning in fuzzy logic and Mamdani inference. Introduction to Natural Language Processing, phases of NLP, parse tree, tokenization, stemming, lemmatization, chunking, Bag of Words and Word2Vec.
S. No.	**Suggested Practicals / Mini Projects**
1	Implement Breadth First Search (BFS) and Depth First Search (DFS) for graph traversal.
2	Implement Uniform Cost Search and Iterative Deepening Search for problem solving.
3	Implement heuristic search techniques including Hill Climbing and A* algorithm.
4	Implement Minimax algorithm for a simple game problem.
5	Implement Alpha-Beta pruning to optimize game tree search.
6	Develop a rule-based expert system using propositional or first-order logic.
7	Implement Bayesian classification or probabilistic reasoning using a suitable dataset.
8	Implement Decision Tree and Naive Bayes classifiers for classification problems.
9	Implement Support Vector Machine (SVM) for supervised learning.
10	Implement K-Means clustering and evaluate clustering performance.
11	Implement a basic neural network using Python for classification or prediction.
12	Perform basic NLP operations such as tokenization, stemming, lemmatization, Bag of Words and Word2Vec.
13	Mini Project: Develop an AI/ML-based application using search, classification, clustering, neural network or NLP techniques.

CLO-PLO Mapping Matrix

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

CLO2	3	3	2	2	1	0	0	0	0	0	0	1	2
CLO3	3	3	3	2	2	0	0	0	0	0	0	1	2.33
CLO4	3	2	2	1	1	0	0	0	0	0	0	1	1.67
CLO5	3	3	2	3	2	1	1	0	0	0	0	2	2.13
Avg PLO	3.0	2.6	2.3	2.0	1.5	1.0	1.0	0	0	0	0	1.2	2.0

Suggested Reading

1	*Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Pearson.*
2	*Elaine Rich, Kevin Knight and Shivashankar B. Nair, Artificial Intelligence, McGraw Hill.*
3	*Tom M. Mitchell, Machine Learning, McGraw Hill.*
4	*Ethem Alpaydin, Introduction to Machine Learning, MIT Press.*
5	*Prateek Joshi, Artificial Intelligence with Python, Packt Publishing.*
6	*S. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications, Prentice Hall of India.*

Teaching-Learning Strategies

Interactive lectures to explain AI concepts, search techniques, reasoning methods, and ML fundamentals.

Demonstrations using Python-based tools for classification, clustering, neural networks, and NLP tasks.

Assignments, case studies, and mini-projects to strengthen practical understanding of AI and ML applications.

Course Code	BCCSLIP526				Semester: 5th	
Course Title	IoT & Smart System Prototyping				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	0	0	4	2	0	50

Course Learning Outcomes (CLOs)	
CLO1	Understand the fundamentals of the Internet of Things (IoT) and smart systems.
CLO2	Design and develop IoT-based solutions using sensors, actuators, and embedded platforms.
CLO3	Explore communication protocols and cloud integration for IoT applications.
CLO4	Develop smart systems for real-world applications such as smart homes, healthcare, agriculture, and industry.
CLO5	Gain hands-on experience in IoT prototyping and deployment.

Suggested Contents/Practicals / Mini Projects/ Hands-on-experiments	
1	**Introduction to IoT and Smart Systems** Evolution and Vision of IoT, Characteristics and Components of IoT, IoT Architecture and Design Principles; Smart Systems: Concepts and Applications, IoT Reference Models, Sensors, Actuators, and Embedded Devices, IoT Hardware Platforms (Arduino, ESP32, Raspberry Pi) Challenges and Future Trends in IoT
2	**Embedded Systems and IoT Prototyping:** Introduction to Embedded Systems: Microcontrollers and Single Board Computers, Arduino Programming Fundamentals, ESP8266/ESP32 Architecture and Programming, Sensor Interfacing Techniques, Actuator Control Mechanisms, Data Acquisition and Signal Conditioning, Rapid Prototyping Techniques
3	**IoT Communication and Networking:** IoT Communication Models, Wireless Sensor Networks, IPv6 for IoT, MQTT Protocol, CoAP Protocol, HTTP and RESTful APIs, ZigBee, Bluetooth Low Energy (BLE), Edge and Fog Computing Concepts

4	**Cloud Integration and Smart Applications:** IoT Cloud Platforms, Device-to-Cloud Communication, Data Storage and Analytics, IoT Dashboards and Visualization, Cloud Services for IoT, Smart Home Systems, Smart Healthcare Systems, Smart Agriculture and Smart City Applications, Industrial IoT (IIoT)
5	**IoT Security, AI Integration and Case Studies:** Security Requirements in IoT, Authentication and Access Control, Secure Communication Protocols, Privacy Issues in IoT, Artificial Intelligence and Machine Learning for IoT, TinyML Concepts, Digital Twins, Case Studies of Smart Systems, Emerging Trends in IoT and Smart Technologies

Note: This is only a suggested list of contents/practicals/lab exercises. Instructors may frame additional exercises relevant to the course and contemporary application domains.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	0	0	1	0	0	0	0	0	0	1	1.75
CLO2	3	3	3	2	3	1	1	0	1	0	1	2	2
CLO3	3	3	2	2	3	1	1	0	0	0	0	2	2.13
CLO4	3	3	3	3	3	3	2	1	1	1	1	2	2.17
CLO5	3	2	3	3	3	1	1	0	2	1	1	3	2.09
Avg PLO	3.00	2.6	2.8	2.5	2.6	1.5	1.3	1.0	1.3	1.0	1.0	2.0	2.0

Suggested Reading

1	*Adrian McEwen, Hakim Cassimally, Designing the Internet of Things, Wiley*
2	*Alice James, Avishkar Seth, Subhas Chandra Mukhopadhyay, "IoT System Design: Project Based Approach",Springer*
3	*Milan Milenkovic, "Internet of Things: Concepts and System Design", Springer*
4	*G. R. Kanagachidambaresan, "Internet of Things Using Single Board Computers: Principles of IoT and Python Programming", Apress*
5	*Haider Raad, "Fundamentals of IoT and Wearable Technology Design", Wiley-IEEE Press*

Teaching-Learning Strategies

Hands-on Laboratory Sessions using modern computing tools and development platforms.
Project-Based Learning through collaborative software development and documentation.
Experiential Learning using industry-standard tools and real-world applications.
Seminar and Presentation on emerging IoT and Smart technologies and industry trends.
Collaborative and Peer Learning through teamwork, code review, and technical discussions.

6th Semester

Course Code	BCCSCCN626				6th Semester	
Course Title	Cryptography & Network Security				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 fundamental concepts of cryptography, classical encryption techniques, and the mathematical principles underlying modern cryptographic systems.
CLO2	Apply symmetric-key cryptographic algorithms and block cipher operation modes to ensure confidentiality in secure communication.
CLO3	Analyze public-key cryptographic algorithms, key management techniques, and secure key exchange protocols for secure communication.
CLO4	Evaluate cryptographic hash functions, message authentication mechanisms, digital signatures, authentication protocols, and public key infrastructure (PKI) for ensuring data integrity, authentication, and non-repudiation.
CLO5	Apply secure communication protocols, including SSL/TLS, IPSec, PGP, and S/MIME, to design secure network communication systems.

Syllabus

Units	Content
1	**Introduction to Cryptography and Mathematical Foundations** Information Security Concepts: CIA Triad, Security Services and Security Mechanisms. Introduction to Cryptography, Classical Encryption Techniques, Substitution Ciphers, Transposition Ciphers, Caesar Cipher, Playfair Cipher, Hill Cipher, Vigenère Cipher, Cryptanalysis and Brute Force Attack.
2	**Symmetric-Key Cryptography** Principles of Symmetric-Key Cryptography, Stream Ciphers, Block Ciphers, Data Encryption Standard (DES), Advanced Encryption Standard (AES), International Data Encryption Algorithm (IDEA), Modes of Block Cipher Operation (ECB, CBC, CFB, OFB and CTR), Random Bit Generation and RC4.

3	**Public-Key Cryptography and Key Management** Principles of Public-Key Cryptography, RSA Algorithm, ElGamal Cryptosystem, Elliptic Curve Cryptography (ECC), Homomorphic Encryption, Secret Sharing Schemes, Key Distribution, Key Exchange Protocols, Diffie–Hellman Key Exchange and Man-in-the-Middle Attack.
4	**Cryptographic Hashing, Authentication and Digital Signatures** Requirements and Security of Hash Functions, Message Digest (MD5), Secure Hash Algorithm (SHA), Birthday Attack, Hash-based Message Authentication Code (HMAC), Authentication Requirements, Authentication Functions, Message Authentication Codes (MAC), Authentication Protocols, Digital Signature Standard (DSS), RSA Digital Signature, ElGamal Digital Signature, Kerberos Authentication Service, X.509 Authentication Service and Public Key Infrastructure (PKI).
5	**Network Security Protocols and Secure Applications** Transport Layer Security Concepts, Secure Socket Layer (SSL), Transport Layer Security (TLS), IP Security (IPSec), IPSec Architecture, Encapsulating Security Payload (ESP), Electronic Mail Security, Pretty Good Privacy (PGP), Secure/Multipurpose Internet Mail Extensions (S/MIME), Secure Electronic Transaction (SET) and Emerging Trends in Cryptography and Network Security.

S. No.	Suggested Practicals / Mini Projects
1	Implementation of classical substitution cipher techniques.
2	Implementation of classical transposition cipher techniques.
3	Implementation of DES and AES encryption algorithms.
4	Implementation of RSA encryption and decryption.
5	Implementation of Diffie–Hellman Key Exchange algorithm.
6	Generation of message digests using MD5 and SHA algorithms.
7	Comparative analysis of classical, symmetric, and asymmetric cryptographic algorithms based on security and performance.

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	2	1	0	2	0	1	0	1	1.57
CLO2	3	3	2	2	3	2	0	2	0	1	0	1	2.11
CLO3	3	3	2	3	3	2	0	2	0	1	0	2	2.10
CLO4	3	3	3	3	3	2	0	3	0	2	0	2	2.40

CLO5	3	3	3	2	3	2	0	2	1	2	0	2	2.30
Avg PLO	3.0	2.8	2.2	2.2	2.8	1.8	0	2.2	1.0	1.4	0	1.6	2.10

Suggested Reading

1	*Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition*
2	*Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition.*
3	*Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.*
4	*Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition*
5	*Information Security, Principles, and Practice: Mark Stamp, Wiley India.*
6	*Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH*

Teaching-Learning Strategies

Interactive Lectures: Use multimedia presentations and real-world case studies to explain complex cryptographic principles, security protocols (SSL/TLS, IPSec), and mathematical foundations.

Problem-Solving Sessions: Conduct dedicated sessions for solving numerical problems related to classical ciphers, RSA, Diffie-Hellman key exchange, and hash function calculations.

Hands-on Laboratory Work: Utilize tools (e.g., OpenSSL, Wireshark, or custom scripts in Python/C++) to implement and test symmetric/asymmetric algorithms, message digests, and network security protocols.

Collaborative Projects: Assign group projects requiring the design and analysis of secure communication systems, including simulated attacks and defense mechanisms.

Demonstrations: Provide live demonstrations of cryptographic attacks (e.g., Brute Force, Birthday Attack) and security protocol handshakes to deepen practical understanding.

Course Code	BCCSCWM626				6th Semester	
Course Title	Wireless and mobile Network security				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	1	0	4	100	NIL

Course Learning Outcomes (CLOs)

CLO1	Explain the fundamentals of wireless and mobile networks and their security requirements.
CLO2	Analyze wireless security protocols, authentication mechanisms, and common wireless attacks.
CLO3	Evaluate security mechanisms for mobile devices, mobile applications, and secure mobile communications.
CLO4	Apply wireless monitoring, intrusion detection, forensic, and incident response techniques to secure wireless environments.
CLO5	Assess emerging technologies such as IoT, 5G, cloud-based wireless systems, and Zero Trust architectures to recommend appropriate security solutions

Syllabus

Units	Content
1	**Fundamentals of Wireless and Mobile Networks** Introduction to wireless communication, Wireless network architectures, IEEE 802.11 WLAN standards, Wireless technologies (Wi-Fi, Bluetooth, ZigBee, NFC, RFID), Cellular networks (2G,3G, 4G LTE, 5G), Wireless security principles, Security challenges in wireless and mobile networks.
2	**Wireless Network Security** Wi-Fi security protocols (WEP, WPA, WPA2, WPA3), Authentication and access control mechanisms, Wireless encryption techniques, Wireless attacks and vulnerabilities, Secure wireless network configuration, Wireless network monitoring and protection.
3	**Mobile Device and Application Security** Mobile operating system security, Mobile device management, Mobile application security, Mobile malware and threats, Secure mobile communication, Mobile authentication and access control, Mobile security best practices

4	**Wireless Monitoring and Incident Response** Wireless traffic analysis, Packet capture and network monitoring, Wireless intrusion detection and prevention systems, Bluetooth, NFC and RFID security, Mobile forensics fundamentals, Wireless incident response and mitigation.
5	**Emerging Trends in Wireless Security** IoT security, 5G security architecture, Cloud and edge security, Zero Trust and BYOD security, AI-driven wireless security, Enterprise wireless security, Future trends in wireless and mobile security.

CLO-PLO Mapping Matrix

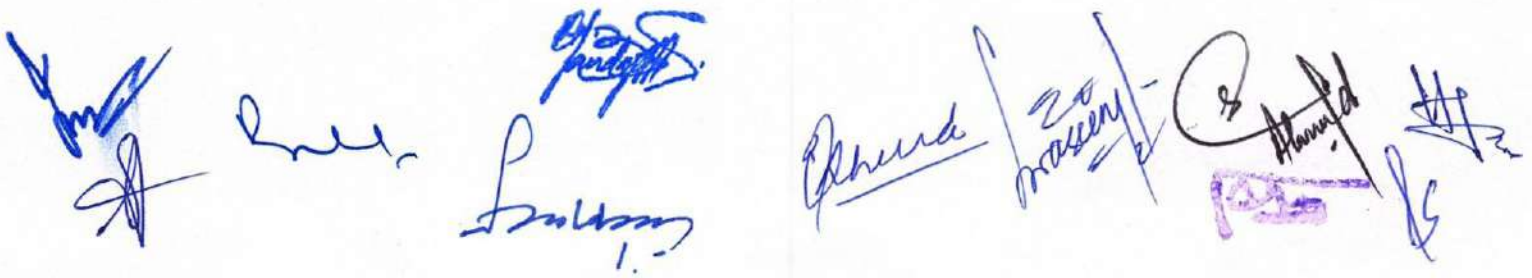
CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	1	1	2	2	0	1	0	1	0	1	1.75
CLO2	3	3	2	2	3	2	0	2	0	1	0	1	2.11
CLO3	3	3	3	2	3	2	0	2	0	1	0	2	2.10
CLO4	3	3	3	3	3	2	0	2	1	2	0	2	2.40
CLO5	3	3	3	3	3	3	1	2	1	2	0	3	2.45
Avg PLO	3.0	2.8	2.4	2.2	2.8	2.2	1.0	1.8	1.0	1.4	0	1.8	2.16

Suggested Reading

1	*William Stallings, Wireless Communications and Networks, 2nd Edition, Pearson.*
2	*Nishith Pathak, Mobile and Wireless Network Security and Privacy, CRC Press.*
3	*Nikos Komninos, Wireless Security and Privacy, CRC Press.*
4	*Theodore S. Rappaport, Wireless Communications: Principles and Practice, 2nd Edition, Pearson*
5	*William C. Y. Lee, Mobile Cellular Telecommunications: Analog and Digital Systems, McGraw-Hill*

Teaching-Learning Strategies

Interactive lectures and classroom discussions to explain wireless security principles, protocols, and emerging technologies like 5G, IoT, and Zero Trust.
Hands-on laboratory sessions using network simulation tools (e.g., Cisco Packet Tracer, NS3) and security analysis tools (e.g., Wireshark) for wireless traffic analysis, monitoring, and intrusion detection.
Problem-solving tutorials and assignments focusing on wireless network configurations, vulnerability assessment, and threat modeling.
Project-based learning where students design and implement secure wireless network architectures or mobile security solutions.
Case studies and seminars on real-world wireless/mobile security incidents, privacy concerns, and emerging industry trends.



Course Code	BCCSCSD626				6th Semester	
Course Title	Secure Software Development				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	2	0	2	3	100	100

Course Learning Outcomes (CLOs)	
CLO1	Explain the principles of secure software development, Secure SDLC, software vulnerabilities, and security design principles for developing secure software systems.
CLO2	Analyze software security requirements, threat models, and secure software architectures to identify and mitigate security risks during software design
CLO3	Apply secure coding principles, authentication, authorization, input validation, and secure programming techniques to develop resilient software applications.
CLO4	Evaluate software security using secure code review, software security testing techniques, and DevSecOps practices throughout the software development lifecycle.
CLO5	Apply secure deployment, software security governance, compliance frameworks, and maintenance practices to enhance the overall security and resilience of software systems.

Syllabus	
Units	Content
1	**Fundamentals of Secure Software Development:** Introduction to Software Security, Security Principles (Confidentiality, Integrity, Availability, Least Privilege, Defense in Depth, Secure by Design), Common Software Vulnerabilities, Secure Software Development Life Cycle (Secure SDLC), Threats, Risks, and Attack Surface, Introduction to Threat Modeling, OWASP Top 10 Overview, Secure Coding Standards and Best Practices.
2	**Secure Requirements Engineering and Software Design:** Security Requirements Engineering, Functional and Non-functional Security Requirements, Abuse Cases and Misuse Cases, Threat Modeling Techniques (STRIDE, DREAD), Secure Software Architecture, Security Design Principles, Secure Design Patterns, Architectural Risk Analysis, Input Validation and Error Handling Strategies.

3	**Secure Coding Practices:** Secure Programming Principles, Secure Coding Guidelines, Input Validation, Output Encoding, Authentication and Authorization, Secure Session Management, Secure Password Storage, Secure File Handling, Exception Handling, Memory Management Issues, Buffer Overflow, Integer Overflow, Secure API Usage and Defensive Programming Techniques.
4	**Software Security Testing and DevSecOps:** Code Review Techniques, Static Application Security Testing (SAST), Dynamic Application Security Testing (DAST), Software Composition Analysis (SCA), Vulnerability Management during SDLC, Unit Security Testing, Integration Security Testing, Fuzz Testing, Continuous Integration/Continuous Deployment (CI/CD), Introduction to DevSecOps.
5	**Software Security Governance and Emerging Trends:** Secure Software Deployment, Secure Configuration Management, Patch and Update Management, Software Supply Chain Security, Software Bill of Materials (SBOM), Security Policies and Compliance, Secure Software Maintenance, Incident Response in SDLC, NIST Secure Software Development Framework (SSDF), OWASP SAMM, Emerging Trends in Secure Software Development.
S. No.	**Suggested Practicals / Mini Projects**
1	• Study the phases of the Secure Software Development Life Cycle (Secure SDLC) and identify common software vulnerabilities using OWASP Top 10 examples. • Perform threat modeling of a simple software application using the STRIDE methodology.
2	• Prepare misuse cases and abuse cases for a sample software application and identify corresponding security requirements. • Perform architectural risk analysis of a software system and recommend appropriate secure design principles and mitigation strategies.
3	• Implement secure coding practices including input validation, authentication, authorization, password hashing, secure session management, and exception handling. •Identify and remediate common coding vulnerabilities using secure programming techniques and code review.
4	• Perform static application security testing (SAST) using SonarQube or SpotBugs. • Perform dynamic application security testing (DAST) using OWASP ZAP and analyze the identified security issues.
5	• Develop a mini project by applying secure software development practices, including secure coding, software security testing, and secure deployment techniques.

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	2	0	2	0	1	0	1	1.78

CLO2	3	3	3	2	3	2	0	2	0	1	0	2	2.10
CLO3	3	3	3	2	3	2	0	2	1	1	0	2	2.20
CLO4	3	3	3	3	3	2	0	2	1	2	0	2	2.40
CLO5	3	3	3	2	3	3	0	3	1	2	1	2	2.33
Avg PLO	3.0	2.8	2.8	2.0	2.8	2.2	0	2.2	1.0	1.4	1.0	1.8	2.16

Suggested Reading

1	*Jason Grembi, Secure Software Development: A Security Programmer's Guide, Cengage Learning, 2008.*
2	*Mark Graff and Kenneth R. van Wyk, Secure Coding: Principles and Practices, O'Reilly Media, 2003.*
3	*Gary McGraw, Software Security: Building Security In, Addison-Wesley Professional, 2006.*
4	*Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw and Nancy R. Mead, Software Security Engineering: A Guide for Project Managers, Addison-Wesley Professional, 2008.*
5	*Neil Madden, API Security in Action, Manning Publications, 2020.*

Teaching-Learning Strategies

Interactive lectures supported by case studies and real-world examples to explain principles of secure software development and threat modeling. Hands-on laboratory sessions focusing on vulnerability assessment, static (SAST) and dynamic (DAST) analysis using tools like SonarQube and OWASP ZAP. Problem-solving sessions addressing secure coding practices, input validation, and mitigation of common vulnerabilities. Collaborative mini-projects to design, develop, and secure software applications, emphasizing DevSecOps practices and secure deployment.

Course Code	BCCSCEH626				6th Semester	
Course Title	Ethical Hacking				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 ethical hacking methodologies, cyber laws, reconnaissance techniques, and information-gathering methods used in cybersecurity.
CLO2	Apply network scanning, enumeration, password auditing, and system security techniques to identify common vulnerabilities
CLO3	Analyze network and web application attacks and recommend suitable defensive measures against common cyber threats.
CLO4	Evaluate security challenges associated with operating systems, wireless networks, mobile devices, cloud environments, and IoT systems.
CLO5	Apply basic security assessment techniques and recommend appropriate security controls to enhance the security posture of computing systems.

Syllabus

Units	Content
1	**Introduction to Ethical Hacking and Information Gathering** Introduction to Ethical Hacking, Hacker Types, Cyber Attack Lifecycle, Ethical Hacking Methodology, Cyber Laws and Ethics, Threats, Vulnerabilities, and Attack Vectors, Footprinting Concepts, Open-Source Intelligence (OSINT), WHOIS, DNS Enumeration.
2	**Network Scanning and System Hacking** Network Scanning Concepts, Host Discovery, Port Scanning, Service Enumeration, Vulnerability Scanning, Password Attacks, Authentication Mechanisms, Privilege Escalation Concepts, Malware Fundamentals, Trojans, Backdoors, and Countermeasures.
3	**Network and Web Application Hacking** Network Sniffing, Spoofing Techniques, Session Hijacking Concepts, Wireless Network Security, Web Application Security Fundamentals, SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), File Inclusion Attacks and Web Application Security Best Practices.

4	**Mobile, Cloud and IoT Security** Windows and Linux Security Fundamentals, Mobile Device Security, Android Security Concepts, Cloud Security Basics, IoT Security Challenges, Email Security, Secure System Configuration, and Patch Management.
5	**Security Assessment and Defensive Strategies** Security Assessment Concepts, Vulnerability Assessment Fundamentals, Security Auditing, Log Analysis, Incident Response Process, Security Policies, Risk Mitigation Strategies, Cyber Threat Intelligence: Fundamentals and Emerging Trends in Ethical Hacking.
S. No.	**Suggested Practicals / Mini Projects**
1	Perform information gathering and footprinting using WHOIS, DNS lookup, NSLookup, traceroute, and OSINT tools. Perform website reconnaissance using Google Dorking techniques.
2	Perform host discovery, port scanning and service enumeration using Nmap. Demonstrate password auditing and password strength analysis using suitable security tools in a controlled laboratory environment.
3	Capture and analyze network traffic using Wireshark. Demonstrate common web application vulnerabilities such as SQL Injection and Cross-Site Scripting (XSS) using DVWA or OWASP Juice Shop.
4	Perform basic Linux/Windows system security configuration, including user account management, password policies and file permissions.
5	Analyze system and network logs to identify suspicious activities. Mini Project: Conduct a basic security audit of a laboratory system or web application and recommend suitable security controls and best practices.

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	2	2	0	3	0	1	0	1	1.78
CLO2	3	3	2	2	3	2	0	2	0	1	0	2	2.00
CLO3	3	3	3	3	3	2	0	2	1	2	0	2	2.33
CLO4	3	3	3	3	3	2	1	2	1	2	0	2	2.27
CLO5	3	3	3	3	3	3	0	3	1	2	1	2	2.27

Avg PLO	3.0	2.8	2.4	2.4	2.8	2.2	1.0	2.4	1.0	1.6	1.0	1.8	2.13

Suggested Reading

1	*Michael T. Simpson, Kent Backman and James Corley, Hands-On Ethical Hacking and Network Defense, 4th Edition, Cengage Learning, 2021.*
2	*Rafay Baloch, Ethical Hacking and Penetration Testing Guide, CRC Press, 2017.*
3	*Georgia Weidman, Penetration Testing: A Hands-on Introduction to Hacking, No Starch Press, 2014.*
4	*Jon Erickson, Hacking: The Art of Exploitation, 2nd Edition, No Starch Press, 2008.*
5	*Patrick Engebretson, The Basics of Hacking and Penetration Testing, 3rd Edition, Syngress, 2013.*

Teaching-Learning Strategies

Deliver fundamental concepts using interactive lectures, multimedia presentations, and real-world cybersecurity case studies.
Demonstrate ethical hacking tools and penetration testing methodologies in a controlled laboratory environment before students perform the experiments independently.
Conduct guided laboratory exercises using virtualization and sandboxed environments to develop practical skills in vulnerability assessment, scanning, and exploitation.
Encourage students to work in teams on laboratory exercises, security assessments, and a capstone penetration testing project to foster teamwork, critical thinking, and communication skills.

Course Code	BCCSLWP626				6th Semester	
Course Title	Web 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	Design well-structured and responsive web pages using HTML5 and CSS3 to create user-friendly web interfaces.
CLO2	Develop interactive client-side web applications using JavaScript, DOM manipulation, jQuery, and JSON for dynamic user experiences.
CLO3	Apply object-oriented programming concepts in Java and implement database connectivity using JDBC for data-driven applications.
CLO4	Develop server-side web applications using Servlets, JSP, and Web Services, incorporating session management and data exchange mechanisms.
CLO5	Design, integrate, test, and demonstrate a complete full-stack web application by combining front-end, back-end, and database technologies to solve a real-world problem.

Syllabus

Units	Content
1	Explore HTML5 Fundamentals: Create simple web pages using headings, paragraphs, text formatting, lists, hyperlinks, images, tables, and semantic HTML5 elements.
2	Design Web Forms and Page Layouts: Develop web pages using forms, input controls, multimedia, navigation menus, iframes, and HTML5 validation attributes.
3	Apply CSS3 for Web Styling: Experiment with selectors, colors, fonts, backgrounds, borders, margins, padding, box model, positioning, and responsive layouts.
4	Develop Interactive Webpages using JavaScript: Implement variables, operators, control statements, functions, arrays, events, DOM manipulation, and client-side form validation.
5	Build a Responsive Front-End Web Application: Integrate HTML5, CSS3, JavaScript, jQuery, and JSON to create a responsive multi-page website with dynamic features.

6	Implement Core Java Programs: Develop programs demonstrating classes, objects, inheritance, interfaces, exception handling, file handling, collections, and multithreading.
7	Develop Database Applications using JDBC: Connect Java applications to a relational database and perform CRUD operations using JDBC.
8	Develop Dynamic Web Applications using Servlets and JSP: Create applications that process user requests, manage sessions, handle cookies, and interact with databases.
9	Implement XML and JSON Processing: Create, parse, validate, and exchange data using XML and JSON in web applications.
10	Develop Basic Web Service Integration: Consume REST APIs or simple web services to retrieve and display dynamic information within a web application.
11	Develop an Integrated Database-Driven Web Application: Combine front-end technologies, Servlets, JSP, JDBC, XML/JSON, and session management to implement a complete CRUD-based application.
12	**Minor Project:** Design, develop, test, and demonstrate a real-world web application (e.g., Student Management System, Library Management System, Online Examination System, Event Registration System, Hospital Management System, or E-Commerce Portal) integrating all relevant technologies studied during the course.

CLO-PLO Mapping Matrix

CLO/PLO	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8	PLO9	PLO10	PLO11	PLO12	Avg CLO
CLO1	3	2	3	1	3	0	0	0	0	2	0	1	2.14
CLO2	3	3	3	2	3	0	0	0	1	2	0	2	2.38
CLO3	3	3	3	2	3	0	0	0	1	2	0	2	2.38
CLO4	3	3	3	3	3	0	0	0	1	2	0	2	2.50
CLO5	3	3	3	3	3	1	0	1	2	2	1	3	2.27
Avg PLO	3.0	2.8	3.0	2.2	3.0	1.0	0	1.0	1.25	2.0	1.0	2.0	2.33

Suggested Reading

1	*Deitel, P. J., & Deitel, H. M., Internet & World Wide Web: How to Program, 5th Edition, Pearson Education.*
2	*Jon Duckett, HTML and CSS: Design and Build Websites, John Wiley & Sons.*

3	*Jon Duckett, JavaScript and jQuery: Interactive Front-End Web Development, John Wiley & Sons.*
4	*Bryan Basham, Kathy Sierra, & Bert Bates, Head First Servlets and JSP, 2nd Edition, O'Reilly Media.*
5	*Brett McLaughlin & Justin Edelson, Java XML, O'Reilly Media.*

Teaching-Learning Strategies

Demonstrate each experiment through live coding before students perform the practical independently in the laboratory.
Conduct guided hands-on laboratory sessions to develop web applications using HTML5, CSS3, JavaScript, Java, JDBC, servlets, JSP, XML/JSON, and REST APIs.
Encourage experiment-based learning through coding, debugging, testing, and incremental enhancement of web applications.
Promote collaborative laboratory activities, peer code reviews, and mini-projects to strengthen problem-solving, teamwork, and communication skills.
Reinforce learning through a capstone web application project integrating front-end, back-end, database connectivity, and web services.

Course Code	BCCSDSS626				6th Semester	
Course Title	System Security				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)

CLO1	Explain the fundamental concepts of system security, operating system security, and access control mechanisms.
CLO2	Analyze system vulnerabilities and apply appropriate system hardening and protection techniques.
CLO3	Apply security mechanisms to protect systems against malware and software exploitation techniques.
CLO4	Evaluate security challenges and protection mechanisms in mobile, virtualization, cloud, and hardware platforms.
CLO5	Analyze security issues in critical infrastructure and cyber-physical systems and recommend appropriate protection strategies.

Syllabus

Units	Content
1	**Operating System Security** Operating System Security Fundamentals, Windows Security Architecture, Linux Security Architecture, User and Kernel Modes, Access Control Models, File System Security, Process Security, Memory Protection, User Authentication, and Privilege Management.
2	**System Hardening and Endpoint Security** System hardening techniques, secure system configuration, patch management, endpoint security, host-based firewalls, anti-malware systems, endpoint detection and response (EDR), security policies, logging, and auditing.
3	**Malware and System Protection** Malware Fundamentals: Viruses, Worms, Trojans, Spyware, Ransomware, Rootkits, Malware Detection Techniques, Sandboxing, Application Whitelisting, Secure Boot and Trusted Platform Module (TPM).
4	**Mobile, Virtualization and Cloud Platform Security** Android Security, iOS Security, Mobile Application Security, Virtual Machine Security, Container Security, Docker Security, Hypervisor Security, Cloud Platform Security Fundamentals.

5	**Critical Infrastructure and Platform Security** SCADA and Industrial Control System Security, Cyber-Physical Systems, IoT Security Fundamentals, Hardware Security Basics, Supply Chain Security, Side-Channel Attacks, and Zero Trust Architecture.

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	2	1	0	2	0	0	0	1	1.71
CLO2	2	3	2	2	3	1	0	2	0	0	0	2	2.14
CLO3	2	3	3	2	3	1	0	2	1	0	0	2	2.13
CLO4	2	3	2	3	3	2	0	2	0	1	0	2	2.22
CLO5	2	3	2	3	2	2	0	3	0	1	0	2	2.22
Avg PLO	2.20	2.80	2.00	2.50	2.60	1.40	0	2.20	1.00	1.00	0	1.80	2.08

Suggested Reading

1	*Ross J. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, 3rd Edition, Wiley, 2020.*
2	*Wenliang Du, Computer Security: A Hands-on Approach, 3rd Edition, 2023.*
3	*William Stallings and Lawrie Brown, Computer Security: Principles and Practice, 5th Edition, Pearson, 2024.*
4	*Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, 7th Edition, Cengage Learning, 2022.*
5	*Charlie Kaufman, Radia Perlman and Mike Speciner, Network Security: Private Communication in a Public World, 3rd Edition, Pearson.*

Teaching-Learning Strategies

Deliver core concepts through interactive lectures, multimedia presentations, and real-world cybersecurity case studies.
Demonstrate system security mechanisms, secure configuration practices, and security tools using guided laboratory demonstrations and virtual environments.
Engage students in problem-based learning, tutorials, group discussions, and analysis of contemporary system security challenges.
Reinforce learning through case studies, security analysis assignments, and seminar presentations on emerging system security technologies and best practices.

Course Code	BCCSDBT626				6th Semester	
Course Title	Blockchain Technology				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)

CLO1	Explain the fundamental concepts, architecture, cryptographic principles, and applications of Blockchain technology.
CLO2	Analyze blockchain data structures, consensus mechanisms, cryptocurrency fundamentals, and transaction validation processes.
CLO3	Develop and deploy basic smart contracts and decentralized applications (DApps) using Ethereum and Solidity.
CLO4	Evaluate various blockchain platforms and their applications in finance, healthcare, supply chain management, identity management, e-governance, and cybersecurity.
CLO5	Assess blockchain security challenges, smart contract vulnerabilities, privacy issues, scalability limitations, and emerging trends in blockchain technology

Syllabus

Units	Content
1	**Introduction to Blockchain** Introduction to Blockchain, Evolution of Blockchain Technology, Features and Characteristics, Blockchain Architecture, Distributed Ledger Technology, Cryptographic Foundations (Hash Functions and Digital Signatures), Types of Blockchain (Public, Private and Consortium), Applications of Blockchain.
2	**Blockchain Fundamentals** Blocks and Transactions, Blockchain Data Structure, Consensus Mechanisms, Proof of Work (PoW), Proof of Stake (PoS), Mining Process, Bitcoin Fundamentals, Cryptocurrency Basics, Digital Wallets.
3	**Smart Contracts and Ethereum** Introduction to Ethereum, Ethereum Architecture, Ethereum Virtual Machine (EVM), Smart Contracts, Introduction to Solidity, Decentralized Applications (DApps), Advantages and Limitations of Smart Contracts.
4	**Blockchain Platforms and Applications** Overview of Blockchain Platforms (Bitcoin, Ethereum and Hyperledger Fabric), Blockchain Applications in Finance, Healthcare, Supply Chain Management, Identity Management, E-Governance and Cyber Security.

5	**Blockchain Security and Future Trends** Blockchain Security Issues, Common Blockchain Attacks, Smart Contract Security, Privacy in Blockchain, Scalability Challenges, Regulatory Issues and Emerging Trends.

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	2	0	1	0	1	0	2	1.82
CLO2	3	3	2	2	3	1	0	1	0	1	0	2	1.82
CLO3	3	3	3	2	3	1	0	1	1	1	0	2	2.00
CLO4	3	3	3	3	3	2	1	1	1	2	1	2	2.08
CLO5	3	3	3	3	3	2	0	3	1	2	0	3	2.33
Avg PLO	3.0	2.8	2.6	2.2	2.8	1.6	1.0	1.4	1.0	1.4	1.0	2.2	2.01

Suggested Reading

1	*Arshdeep Bahga and Vijay Madisetti, Blockchain Applications: A Hands-On Approach, VPT, 2020.*
2	*Melanie Swan, Blockchain: Blueprint for a New Economy, O'Reilly Media, 2015.*
3	*Imran Bashir, Mastering Blockchain, 4th Edition, Packt Publishing, 2023.*
4	*Vikram Dhillon, David Metcalf and Max Hooper, Blockchain Enabled Applications, 2nd Edition, Apress, 2021.*
5	*Roger Wattenhofer, The Science of the Blockchain, Inverted Forest Publishing, 2017.*

Teaching-Learning Strategies

Interactive lectures explaining blockchain architecture, distributed ledger technology, and cryptographic principles.

Problem-solving sessions focused on consensus mechanisms, mining processes, and transaction validation.

Hands-on tutorials using Ethereum and Solidity for smart contract development and DApp deployment.

Case studies and seminar discussions on blockchain applications in sectors like finance, supply chain, and e-governance.

Course Code	BCCSDCF626				6th Semester	
Course Title	Cyber Forensics				Maximum Marks	
Scheme & Credits	Hours Per Week			Credits	Theory	Practical
	L	T	P			
	3	0	0	3	100	0

Course Learning Outcomes (CLOs)

CLO1	Explain the principles, legal aspects, and methodologies of digital and cyber forensic investigations.
CLO2	Apply appropriate techniques for acquisition, preservation, and analysis of digital evidence from computing systems.
CLO3	Analyze cyber incidents using network, malware, email, and web forensic techniques.
CLO4	Examine digital evidence from mobile devices, cloud environments, social media, and IoT systems.
CLO5	Evaluate advanced forensic challenges, emerging technologies, and prepare comprehensive forensic investigation reports.

Syllabus

Units	Content
1	**Fundamentals of Digital and Cyber Forensics** Introduction to Digital and Cyber Forensics, Cyber Crime Investigation Process, Digital Evidence, Types of Digital Evidence, Chain of Custody, Forensic Investigation Methodology, Legal and Ethical Issues, Cyber Laws, Evidence Documentation, and Reporting.
2	**Digital Evidence Acquisition and Host Forensics** Evidence Acquisition and Preservation, Disk Imaging, Hashing and Integrity Verification, Storage Media Forensics, Windows and Linux File System Forensics, Registry Analysis, Log Analysis, Memory (RAM) Forensics, Deleted File Recovery, Digital Forensic Tools.
3	**Network and Cyber Forensics** Cyber-attack investigation, network traffic analysis, packet capture and analysis, firewall and IDS/IPS log analysis, email forensics, web browser forensics, malware forensics, incident response, and cyber threat investigation.

4	**Mobile, Cloud and IoT Forensics** Mobile Device Forensics, Android and iOS Forensics, SIM Card Analysis, Social Media Forensics, Cloud Forensics, Virtual Machine Forensics, IoT Forensics, Challenges in Modern Digital Investigations.
5	**Advanced Forensics and Emerging Trends** Anti-Forensics Techniques, Fileless Malware Investigation, Ransomware Investigation, Cryptocurrency and Blockchain Forensics, AI-Assisted Digital Forensics, Forensic Reporting, Case Studies, and Emerging Trends.

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	2	2	0	3	0	1	0	1	1.78
CLO2	3	3	2	2	3	2	0	2	0	1	0	2	2.00
CLO3	3	3	3	3	3	2	0	2	1	2	0	2	2.33
CLO4	3	3	3	3	3	2	0	2	1	2	0	2	2.33
CLO5	3	3	3	3	3	3	0	3	1	3	1	3	2.42
Avg PLO	3.0	2.8	2.4	2.4	2.8	2.2	0	2.4	1.0	1.8	1.0	2.0	2.17

Suggested Reading

1	*Bill Nelson, Amelia Phillips, Christopher Steuart and Frank Enfinger, Guide to Computer*
2	*André Årnes, Digital Forensics, 2nd Edition, Wiley, 2022.*
3	*Eoghan Casey, Handbook of Digital Forensics and Investigation, Academic Press.*
4	*Forensics and Investigations, 7th Edition, Cengage Learning, 2024.*
5	*Nihad A. Hassan, Digital Forensics Basics: A Practical Guide Using Windows OS, Apress, 2019.*
6	*EC-Council, Computer Forensics: Investigating Network Intrusions and Cyber Crime, Latest Edition.*

Teaching-Learning Strategies

Interactive Lectures: Focusing on forensic principles, legal frameworks, and emerging cyber threats.

Hands-on Laboratory Sessions: Using tools like Autopsy, FTK Imager, Volatility Framework, and Wireshark for evidence acquisition, analysis, and recovery.
Case Study Analysis: Investigating real-world cyber incidents and security breaches.
Collaborative Projects: Mini-projects involving fully digital forensic investigations, report preparation, and evidence chain of custody documentation.