Faculty of Science, Technology and Architecture (FoSTA)

School of Computer Science and Engineering

Department of Computer and Communication Engineering

III Semester Syllabus

CCE2101 Digital Design and Computer Architecture **[3 1 0 4]**

Basic Structure of Computers: Computer Types, Functional Units, Basic Operational Concepts, Software, Performance; Digital logic circuits: logic gates, Boolean algebra, map simplification, combinational circuits, flip-flops, sequential circuits; Digital Components: Integrated circuits, decoders, multiplexers, registers, shift registers, binary counters; Machine instructions and programs: numbers, arithmetic operations and characters, memory locations and addresses, instructions and instruction sequencing, addressing modes, assembly language, additional instructions, encoding of machine instructions; micro-operation and their RTL specification Arithmetic: addition and subtraction of signed numbers, design of fast adders, multiplication of positive numbers, signed operand multiplication, fast multiplication, integer division, floating point numbers and operations; Introduction to input/output processing: programmed controlled i/o transfer, interrupt controlled I/O transfer, DMA controller; Pipelining and pipeline hazards: design issues of pipeline architecture; Instruction level parallelism and advanced issues.

References:

1. M. M. Mano, Computer System Architecture, (3e), Pearson Education, 2014.

2. C. Hamacher, Z. Vranesic, S. Zaky, Computer Organization, (6e), McGraw Hill, 2011.

3. J. P. Hayes, Computer Architecture and Organization, (3e), McGraw Hill, 2017.

4. T. L. Floyd, Digital Fundamentals, (10e), Pearson Education, 2014.

5. W. Stallings, Computer Organization and Architecture–Designing for Performance, (8e), Pearson Education, 2010.

CCE2102 Data Communication and Computer Networks **[3 1 0 4]**

Introduction: Computer networks and distributed systems, Classifications of computer networks, Preliminaries of layered network structures. Data communication Components: Representation of data and its flow, Various connection topologies, Protocols and Standards, OSI model, Transmission Media. Network Topology and Bandwidth: Wired LAN, Wireless LAN, Virtual LAN. Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum, Data Link Layer and Medium Access Sublayer: Fundamentals of Error Detection and Error Correction, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go-back–N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA. Network Layer: Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols. Transport Layer: Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Congestion Control; Quality of Service (QoS), QoS improving techniques - Leaky Bucket and Token Bucket algorithms. Application Layer: DNS, DDNS, TELNET, EMAIL, FTP, WWW, HTTP, SNMP, Bluetooth, Firewalls.

Reference Books:

1. A S Tanenbaum,“Computer Networks”, 6th Edition, Pearson, 2021.
2. Behrouz A. Forouzan, “Data Communications and Networking with TCP/IP Protocol Suite”, 6th Edition. MC Graw Hill, 2021.
3. William Stallings, “Data and Computer Communications”, 10th edition, Pearson Education, 2017

CCE2103 Data Structures and Algorithms **[3 1 0 4]**

Introduction: algorithm specification; Performance analysis: time and space complexity, asymptotic notation; Array: types, implementation, operations, sorting and searching techniques; Linked List: types, implementation and operations; Stack: implementations using array and linked list, operations and its applications (arithmetic expression conversions and evaluations, recursion, tower of Hanoi, merge sort, quick sort, etc); Queue: types, implementations using array and linked list, operations and its applications; Tree: terminologies, different types, representation of binary tree using array and linked list, binary search tree, heap, heap sort, priority queue, AVL trees, B-tree; Graphs: Basic definitions for graph, adjacency matrix, breadth-first search, depth-first search.

Reference

1. Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein, “Data Structures using C”, Pearson Education, 2013.
2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, “Fundamentals of Data Structures in C”, University Press (India) Pvt. Ltd., 2014.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, “Data Structures and Algorithms”, Pearson Education, 2012.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to algorithms”, PHI, Third Edition, 2009.
5. Seymour Lipschutz, “Data Structures with C (Schaum's Outline Series)”, McGraw Hill Education Private Limited, 2011.
6. Mark Allen Weiss, “Data structures and Algorithm Analysis in C”, Pearson, Second edition, 2014.

Reference

1. Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein, “Data Structures using C”, Pearson Education, 2013.
2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, “Fundamentals of Data Structures in C”, University Press (India) Pvt. Ltd., 2014.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, “Data Structures and Algorithms”, Pearson Education, 2012.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, “ Introduction to algorithms”, PHI, Third Edition, 2009.
5. Seymour Lipschutz, “Data Structures with C (Schaum's Outline Series)”, McGraw Hill Education Private Limited, 2011.
6. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Pearson, Second edition, 2014.

CCE2120 Object-Oriented Programming using C++ **[3 1 0 4]**

Introduction to Object-Oriented Programming Concepts: need of object-oriented programming, difference between procedural and object oriented language, Characteristics of object oriented programming; C++ Fundamentals: History, Evaluation of C and C++, Revision of C Concepts, Difference between C and C++, understanding of C++ as the superset of C; Headers files used in C++, Namespaces, C++ keywords, References, Dynamic Allocation Operators, Data types: basic, derived and User Defined Data Types in C++; Operators in C++: types of operators, insertion and exertion operators, scope resolution operat.or, member access operator, Type casting, Operator Overloading in C++; Arrays and Strings: 1-D array, 2-D array, Operations on arrays, Strings and character arrays, Operations on character arrays and Strings; Functions: function declaration, function definition, function calling; Recursive functions; Passing arguments, Returning values; Objects and Classes: defining classes, object creation, access specifiers, Constructors and its types; Inheritance: Need of Inheritance, Types of Inheritance-Single, multilevel, multiple, hybrid inheritance, virtual keyword, virtual function, Virtual base class, abstract class, This pointer; Polymorphism: compile time and runtime polymorphism.

Reference

1. C++: The Complete Reference, Herbert Schildt, McGraw-Hill/Osborne
2. Learn To Program with C++, John Smiley, McGraw-Hill Education.
3. C++ Primer (5th Edition), Josée Lajoie and Stanley B. Lippman, Addison-Wesley.
4. The C++ Programming Language, Bjarne Stroustrup, Pearson.

CCE2121 Object-Oriented Programming using JAVA **[3 1 0 4]**

Introduction: history and evolution of OOP, Introduction to OOPS and classes: class and object fundamentals, introduction to methods/functions, introduction to JDK, JRF and JVM, variables and data types, Unicode system, naming conventions, object initialization constructors, Singleton Class and its Applications and garbage collection, this keyword, overloading, objects as parameters, argument passing, returning objects, recursion, access control, classes within classes, string class; I/O basics: reading console input, writing console output, files; Inheritance: basics, multilevel hierarchy, overriding, abstract classes; Packages and Interfaces; exception handling; Multithreaded programming; String handling; Generic Classes: The collection framework, array list and vector, sets, map; Database programming using JDBC; Java Server Technologies: servlet, java Server Pages, Introduction and implementation of Remote Method Invocation.

References:

1. H. Schildt, Java: The Complete Reference, (10e), McGraw Hill, 2017.
2. C. Horstmann, Core Java Volume-1 Fundamentals, (11e), Prentice Hall of India, 2020.
3. S. Holzner, Java 8 programming Black Book, (1e), Dream Tech, 2015.
4. P. Deitel, H. Deitel, Java How to Program, (11e), Pearson Education, 2018.
5. E. Balagurusamy, Programming with Java: A Primer, (5e), McGraw Hill, 2017

CCE2130 Data Communication and Computer Networks Lab **[0 0 2 1]**

Experiment with Packet Tracer: Introduction to Packet tracer and networking device components; Router Mode, Switch/Router basic commands; designing of star topology using HUB and Switch, IP configuration of end devices; configuring DHCP server, static routing, RIP, OSPF, VLAN and NAT; Network programming: Transmission Control Protocol (TCP) socket and User Datagram Protocol (UDP) socket; Network Utilities: PING, NETSTAT, IPCONFIG, IFCONFIG, ARP, TRACE-ROUTE

Reference:

1. A S Tanenbaum,"Computer Networks", 6th Edition, Pearson, 2021.
2. Behrouz A. Forouzan, "Data Communications and Networking with TCP/IP Protocol Suite", 6th Edition. MC Graw Hill, 2021.
3. William Stallings, "Data and Computer Communications", 10th edition, Pearson Education, 2017.

CCE2131 Data Structures and Algorithms Lab **[0 0 2 1]**

C revision: functions, arrays, pointers, dynamic memory allocation, passing arrays to functions through pointers, structures, arrays of structures, passing structures to functions, linear search, bubble sort, insertion sort, selection sort. Array: insertion, deletion, matrix operation. Linked List: singly, doubly, and circular linked lists: inserting, deleting, and reversing a linked list, Stacks and Queues: implementation of stacks & queues using array and linked lists, conversion of infix to postfix and evaluation of postfix expressions. Trees: traversal of Trees, BST and AVL tree implementation, quick sort, heap sort, merge sort. Graph: Implementation of breadth-first search, depth-first search.

Reference

1. Aaron M. Tenenbaum, Yedidyah Langsam, Moshe J. Augenstein, "Data Structures using C", Pearson Education, 2013.
2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, "Fundamentals of Data Structures in C", University Press (India) Pvt. Ltd., 2014.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, 2012.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, " Introduction to algorithms", PHI, Third Edition, 2009.
5. Seymour Lipschutz, "Data Structures with C (Schaum's Outline Series)", McGraw Hill Education Private Limited, 2011.
6. Mark Allen Weiss, "Data structures and Algorithm Analysis in C", Pearson, Second edition, 2014.

CCE2170 Project-based Learning I **[0 0 2 1]**

Introduction to different sensors and actuators like water level detector, temperature sensor, resistive transducers, proximity sensors, humidity sensors, optical sensors, servo motor, DC motor, buzzer, LED, and LED matrix, seven segment display etc. Working principle and range of measurement for these devices.

Introduction and detail pin diagram for general IoT development boards like Arduino and Raspberry pi. Basics of programming for Arduino and interfacing between different sensors and actuators.

Design of basic smart device to measure some physical/electrical/mechanical quantity and reflect its response on actuators like burglar alarm, automatic water supply for water tanks, display of text/digits on screen.

Reference:

1. Bahga ,V. Madisetti, Internet of Things A Hands-on-Approach, (1e), University Press, 2015.
2. F. daCosta, Rethinking the Internet of Things: A Scalable Approach to Connecting Everything, (1e), Apress Publications, 2013 3.
3. Massimo Banzi , Michael Shiloh Getting Started With Arduino: The Open Source Electronics Prototyping Platform 4TH Edition 2022

IV Semester Syllabus

CCE2201 Relational Database Management Systems **[3 1 0 4]**

Introduction: database systems, RDBMS definition, data models, 3-schema architecture, challenges in building RDBMS, different components of a RDBMS. Relational data model: concept of relation and its characteristics, schema-instance, integrity constraints, E/R Model, Extended E/R model, converting the database specification in E/R and Extended E/R notation to the relational schema; Relational Query Language: relational algebra operators - selection, projection, cross product, various types of joins, division, example queries; Introduction to SQL: data definition in SQL, table and different types of constraints definitions, data manipulation in SQL, nested queries, notion of aggregation; Relational Database Design: functional dependencies and normal forms, Armstrong's axioms for FD's, closure of a set of FD's, minimal covers, definitions of 1NF, 2NF, 3NF and BCNF, decompositions and desirable properties of them, algorithms for 3NF and BCNF normalization, multi-valued dependencies and 4NF; Transaction Processing: concepts of transaction processing, ACID properties, concurrency control, locking based protocols, recovery and logging methods; Data Storage and Indexing: file organizations, primary, secondary index structures, hash-based indexing, dynamic hashing techniques, multi-level indexes.

References:

1. Avi Silberschatz, Henry F. Korth, S. Sudarshan, “Database System Concepts”, TMH, New Delhi, 2013
2. R. Elmasri, S. B. Navathe, “Fundamentals of Database Systems”, Addison & Weisely, New Delhi, 2010
3. Ramakrishnan, J. Gehrke, Database Management Systems, (3e), McGraw Hill, 201
4. I. Bayross, SQL, PL/SQL The Programming Language of Oracle, (4e), BPB Publications, 2010.
5. C. J. Date, An Introduction to Database Systems, (8e), Prentice Hall of India, 2006.
6. Ramakrishnan, J. Gehrke, Database Management Systems, (3e), McGraw Hill, 2014.

CCE2202 Operating Systems **[3 1 0 4]**

Introduction: Evolution of operating system, classification of operating system, operating system structure, services, functions, design and implementation, system programs, system calls, virtual machines, system boot; Processes: Concept, process scheduling, operations on processes, inter- process communication; Linux threads: Basic concepts, multithreaded models, thread libraries; CPU scheduling: Scheduling criteria, scheduling algorithms, thread scheduling; Process synchronization: Concept of synchronization, critical section problem, Dekker's algorithm, peterson's solution, synchronization hardware, semaphores, classical problems on synchronization, monitors; Deadlock: Deadlock concept, deadlock characterization, methods for handling deadlock, prevention, avoidance, detection, recovery from deadlock; Memory management: Concept of logical and physical memory, swapping, contiguous memory allocation, paging, page table structure, segmentation, paging combined with segmentation, working of intel- 32/64; Virtual memory management: Demand paging, copy-on write, page replacement, allocation of frames, thrashing, memory mapped files, allocating kernel memory; Files: file concept, access methods, directory structure, file system mounting, file sharing; Disk: architecture, scheduling algorithms; Security problem: Program threats, system and network threats; Case study: Linux / Solaris / Mac / Windows operating system.

References:

1. A. S. Tannenbaum, Modern Operating Systems, (4e), Pearson, 2016.
2. A. Silberschatz, P. B. Galvin, Operating System Concepts, (9e), Wiley, 2018.
3. W. Stallings, Operating Systems: Internals and Design Principles, (9e), Pearson, 2009.
4. H. Sibsankar, A. A. Alex, Operating Systems, (6e), Pearson, 2010.
5. Andrew S. Tanenbaum and Albert S. Woodhull, Operating Systems: Design and Implementation, (3e), Prentice Hall Software Series, 2015.
6. J. A. Harris, Schaum's Outline of Operating Systems, (2e), McGraw-Hill publications, 2002.

CCE2220 Software Engineering **[3 1 0 4]**

Software Engineering: introduction, importance, evaluation, characteristics, and components.Software applications; Software development process models: waterfall model, prototyping model, spiral model, RAD model; agile modelling; Requirement engineering: problem analysis, requirement verification, requirement validation modularity; Software project management: cost estimation, project scheduling, risk management, quality assurance, project monitoring. Estimation techniques: size estimation- LOC estimation, function count, cost estimation, Halstead size estimation, Software design: analysis modeling, functional modeling, behavioral modeling; unified modeling language; Software architecture; Data design: data modeling, data structures; Software testing: white box (unit and integration), black box (system level, egression); Software maintenance: maintenances characteristics, maintainability, maintenances tasks, maintenances side effects; Current trends in software engineering

Reference

1. R. S. Pressman, “Software Engineering: A Practitioners Approach”, McGraw Hill.
2. K.K. Aggarwal and Y. Singh, “Software Engineering”, New Age International Publishers.
3. P. Jalote, “Software Engineering”, Wiley.
4. I. Sommerville, “Software Engineering”, Addison Wesley.

CCE2221 Automata Theory **[3 1 0 4]**

Introduction; Alphabets, Strings and Languages; Automata and Grammars, Deterministic finite Automata (DFA)-Formal Definition, State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata, Myhill-Nerode Theorem. Regular expression (RE), Definition, Operators of regular expression, Kleen's Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non-Regular Languages, Pumping Lemma for regular Languages. Closure properties of Regular Languages, Decision properties of Regular Languages, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA. Context free grammar (CFG) and Context Free Languages (CFL): Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammer, Inherent ambiguity, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure properties of CFLs, Pumping lemma for CFLs.
Push Down Automata (PDA): Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of PDA and CFG, CFG to PDA and PDA to CFG, Two stack PDA Turing machines (TM): Basic model, definition and representation, Instantaneous Description, Language acceptance by TM, Variants of Turing Machine, TM as Computer of Integer functions, Universal TM, Church's Thesis, Recursive and recursively enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs. Post correspondence problem (PCP), Modified PCP.

Reference

1. Hopcroft, Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education.
2. K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science: Automata, Languages and Computation", PHI Learning Private Limited, Delhi India.
3. Peter Linz, "An Introduction to Formal Language and Automata", Narosa Publishing house.
4. Papadimitrou, C. and Lewis, C.L., "Elements of the Theory of Computation", PHI Learning Private Limited, Delhi India.
5. Harry R. Lewis and Christos H. Papadimitriou, Elements of the theory of Computation, Second Edition, Prentice-Hall of India Pvt. Ltd.
6. Micheal Sipser, "Introduction of the Theory and Computation", Thomson Learning.

CCE2240 Foundation of Data Science **[3 0 0 3]**

Basics of Data Science: Introduction, Typology of problems, Importance of linear algebra, statistics and optimization from a data science perspective, Structured thinking for solving data science problems; Linear Algebra: Matrices and their properties (determinants, traces, rank, nullity, etc.), Eigenvalues and eigenvectors, Matrix factorizations, Inner products, Distance measures, Projections, Notion of hyper planes, half-planes; Probability, Statistics and Random Processes: Probability theory and axioms, Random variables, Probability distributions and density functions (uni-variate and multivariate), Expectations and moments, Covariance and correlation, Statistics and sampling distributions, Hypothesis testing of means, proportions, variances and correlations, Confidence (statistical) intervals, Correlation functions, White-noise process; Optimization: Unconstrained optimization, Necessary and sufficiency conditions for optima, Gradient descent methods, Constrained optimization, KKT conditions, Introduction to non-gradient techniques, Introduction to least squares optimization, Optimization view of machine learning; Introduction to Data Science Methods: Linear regression as an exemplar function approximation problem, Linear classification problems.

Reference

1. G. Strang, Introduction to linear algebra, Wellesley, (5e), MA: Wellesley-Cambridge Press, 2016.
2. J. S. Bendat, A. G. Piersol, Random data: analysis and measurement procedures, (4e), John Wiley & Sons.2010.
3. D. C. Montgomery, G. C. Runger, Applied statistics and probability for engineers, (5e), John Wiley & Sons,2011.
4. C. O'Neil, R. Schutt, Doing data science: Straight talk from the frontline, O'Reilly Media, Inc., 2016.

CCE2241 Web Programming **[3 0 0 3]**

Introduction: an overview of internet and “the web”, web system architecture; HTTP: basics of HTTP request and response, HTTP methods, headers, content transport (push and pull), drawbacks HTTP1.0, introduction to HTTP1.1, HTTPS, SSL.
Client side programming: Introduction to HTML, HTML tags: Text formatting, URLs, lists, iframes, object placement, tables, forms, document object model (DOM) and advancements in HTML, Styling with CSS and advancements in CSS; JavaScript: Introduction, variables, primitive and non-primitive data types, operators, control statements, loops, functions, arrays, JSON, JavaScript inbuilt Objects and APIs, event handling, pattern matching and form validation with regular expressions, working with classes, objects, constructors and inheritance; Server-side programming: Three tier model, PHP – constructs, form validation, sessions, session tracking techniques and working with database. XML: Syntax and semantics, document structure, DTDs. Angular JS: Overview, MVC architecture, directives, controllers, modules. Node JS: Modules, NPM modules, create, edit, and publish NPM modules.

Reference

1. J. C. Jackson, Web Technologies: A Computer Science Perspective, Pearson Education, 2007.
2. Head First HTML5 Programming, Eric Freeman, Elisabeth Robson, O'Reilly Media, Inc. 2011.
3. J. Sklar, Web Design Principles, (5e), Cengage, 2015.
4. DT. E. Services, HTML 5 Black Book, (2e), Dreamtech Press, 2016.
5. CSS: The Definitive Guide, 5th Edition by Eric Meyer, Estelle Weyl, O'Reilly Media, Inc. 2023.
6. JavaScript: The Definitive Guide, 7th Edition, by David Flanagan, O'Reilly Media, Inc. 2020.
7. S. Seshadri, Angular: Up and Running- Learning Angular, Step by Step, (1e), Shroff/O'Reilly, 2018.
8. B. M. Harwani, Developing Web Applications in PHP and AJAX, (1e), McGraw Hill, 2010.
9. D. Herron, Node.js Web Development: Server-side development with Node 10 made easy, (4e), Packet Publishing, 2018.

CCE2242 Cryptography and Network Security **[3 0 0 3]**

Introduction: confidentiality, integrity, availability, OSI security architecture; Number theory: finite fields, Galois field, primes, primality testing, factoring algorithms; Probability and information theory: Shannon's theory, perfect security; Classical ciphers; Block ciphers: DES, AES, electronic codebook mode, cipher block chaining mode, cipher feedback mode, output feedback mode, counter mode; Pseudorandom number generation; Stream ciphers; Cryptographic hash functions; Message authentication codes; Public-key cryptography: computational security, computational assumptions, RSA, ElGamal, elliptic curve cryptography, digital signatures, Diffie- Hellman key exchange; Operating systems security: security capabilities of different platforms, identification, authentication, user accounts, file permissions, backups, access control, firewalls, methods of protection, ownership, assessing and securing a system, information warfare, security administration, corporate espionage

REFERENCE

1. W. Stallings, "Cryptography and Network Security, Principles and Practices", 6th Edition, Pearson Education, 2013.
2. B. A. Forouzan, "Cryptography and Network Security", 3rd Edition, McGraw Hill, 2015.
3. D. Stinson, Cryptography: Theory and Practice, (4e), CRC Press, 2018.
4. Pieprzyk, T. Hardjono, J. Seberry, "Fundamentals of Computer Security", springer-Verlag Berlin Heidelberg, 2013.
5. C. P. Pfleeger, "Security in Computing", 4th Edition, Prentice Hall, 2014.

CCE2130 Relational Database Management Systems Lab **[0 0 2 1]**

Drawing ER diagram, Conversion of ER diagram to relational model, Experiments on DDL, DML, DCL, and DQL. Basic SQL, Advanced SQL, Data Integrity Constraints and Built-in Functions, Experiments on Basic PL/SQL, PL/SQL Exceptions and Transactions, PL/SQL Cursors, PL/SQL Procedures, Functions and Packages, Triggers, Views, Indices, Stored Procedures, DB application development with front end.

Reference

1. Avi Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", TMH, New Delhi, 2006
2. R. Elmasri, S. B. Navathe, "Fundamentals of Database Systems", Addison & Weisely, New Delhi, 2008
3. "Teach yourself SQL & PL/SQL using Oracle 8i & 9i with SQLJ", Ivan Bayross, BPB Publications, 2010
4. Avi Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", TMH, New Delhi, 2006

CCE2131 Operating Systems Lab **[0 0 2 1]**

Testing the use of UNIX commands; Working with VI editor; Shell: UNIX shell commands, System Administration: user management, security, file management; Inter-process communication: shared memory, message passing, pipes; UNIX system calls: system calls for process management, file management; Process synchronization: bounded buffer problem, Peterson's solution, semaphore; Building multi-threaded and multi-process applications: multithreading using pthread library; CPU scheduling algorithms; Deadlock: detection algorithms, deadlock avoidance algorithms; Page replacement algorithms; Memory allocation algorithms; Disk scheduling algorithms.

Reference

1. S. Das, "Unix Concepts and Applications", 4th Edition, Tata McGraw-Hill, 2017.
2. A. Silberschatz, P. B. Galvin and G. Gagne, "Operating System Concepts", 9th Edition, Wiley, 2014.
3. R. Blum, and C. Bresnahan, "Linux Command Line and Shell Scripting Bible", 3rd Edition, Wiley, 2015.
4. Maurice J. Bach, "The Design of the UNIX Operating System", Pearson Education.

CCE2270 Project-based Learning II**[0 0 2 1]**

Design of basic smart IoT project with multiple sensors and actuators as a solution to real time problem in security, healthcare, agriculture, or traffic control systems. Data collection and transmission techniques between two or multiple IoT devices with wired or wireless mode of transmission. Transmission with Bluetooth or wi-fi with different wireless sensors like- NOD MCU ESP8266, NOD MCU2102, HC05 Bluetooth module, Bluetooth Transceiver Module with TTL etc. Introduction to different IoT protocols - MQTT, Zigbee, LoRa for power efficient and long- range transmission in online/offline mode. Data Analysis and presentation on website or user application platforms.

Reference

1. Hanes, David, et al. IoT fundamentals: Networking technologies, protocols, and use cases for the internet of things. Cisco Press, 2017.
2. Suzie Boss, Jane Krauss, Leslie Conery-Reinventing Project-Based Learning: Your Field Guide to Real-World Projects in the Digital Age. 2020.
3. Robert M. Capraro, Scott. W. Slough (auth.), Robert M. Capraro, Mary Margaret Capraro, James R. Morgan (eds.) STEM Project-Based Learning: An Integrated Science, Technology, Engineering, and Mathematics (STEM) Approach, 2013
4. Michael Shiloh Getting Started With Arduino: The Open Source Electronics Prototyping Platform 4TH Edition 2022

V Semester Syllabus

CC3101 Design and Analysis of Algorithms [3 1 0 4]

Introduction: Algorithm Definition and Criteria of Algorithms, Iterative and Recursive algorithms, Performance Analysis, Asymptotic Notations, Space Complexity, Time Complexity, Performance measurement of iterative and recursive algorithms, Solving Recurrence Relations: Substitution Method, Iterative Method, Recursive Tree Method, Master Method

Divide and Conquer and Heaps: Introduction, Binary Search, Heap Sort., Tim Sort, Merge Sort, Quick Sort, Randomized Quick Sort, Lower bound on Comparison based Sorting, Sorting in Linear Time: Counting Sort, Radix Sort, Strassen's matrix multiplication.

Greedy Strategy: Introduction, Knapsack Problem, Huffman Coding, Union and Find Operation (Set and Disjoint Set), Minimum Cost Spanning Tree Algorithms (Prim's and Kruskal's, Single Source Shortest Path (Dijkstra's Algorithm)

Dynamic Programming: Introduction, Single Source Shortest Path (Bellman and Ford Algorithm), All Pair Shortest Path (Floyd Warshal's Algorithm), 0/1 Knapsack Problem, Travelling Salesperson Problem, Longest Common Subsequence, Matrix Chain Multiplication.

Graph Search Algorithm: Graph representation, Red Black Tree, Breadth First Search and Depth First Search, Backtracking: Introduction, N-Queens Problem, Graph Colouring and Hamiltonian Cycles, Branch and Bound: Introduction, FIFO and LC Branch and Bound, 0/1 Knapsack Problem, Travelling Salesman Problem

String Matching: Naïve String Matching, Rabin Karp Algorithm, Knuth-Morris-Pratt Algorithm

Complexity Classes: Basic concepts, non-deterministic algorithms, P, NP, NP-Complete, and NP-Hard classes.

References

1. E. Horowitz, S. Sahni, and S. Rajasekaran, *Fundamentals of Computer Algorithms*, 2nd ed. University Press, 2008.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 4th ed. MIT Press, 2022.
3. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, 1st ed. Pearson Education, 1999.
4. J. Kleinberg and É. Tardos, *Algorithm Design*, eTextbook ed. Pearson, 2021.
5. R. Neapolitan, *Foundations of Algorithms*, 5th ed. Jones and Bartlett Publishers, 2014.

CC3102 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING [3 1 0 4]

Artificial Intelligence Concepts: state space representation and search; Heuristic search techniques: hill climbing, best first search, A*, AO*, constraint satisfaction; Knowledge representation and reasoning; Formal logic and unification algorithms; Planning algorithms, goal stack planning, nonlinear planning using constraint posting, hierarchical planning; Case-based reasoning; Optimization algorithms, genetic algorithm, ant colony optimization, particle swarm optimization, simulated annealing;

Supervised Machine Learning Algorithms: classification algorithms – KNN, decision tree, naïve Bayes, support vector machine, regression, random forests.

Unsupervised Machine Learning Algorithms: Principal component analysis, k-means.

Machine Learning Performance Evaluation Metrics: classification accuracy, logarithmic loss, confusion matrix, area under curve, F1 score, mean absolute error, and mean squared error.

References:

1. *S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Pearson Education, 2020.*
2. T. M. Mitchell, *Machine Learning*, 1st ed. McGraw Hill, 2017.
3. D. Simon, *Evolutionary Optimization Algorithms*, 1st ed. Wiley, 2013.
4. D. Khemani, *A First Course in Artificial Intelligence*, 1st ed. McGraw Hill, 2015.
5. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. John Wiley, 2002.

CE3120 Digital Image Processing and Computer Vision [3 1 0 4]

Introduction to Digital Image Processing: Fundamentals, Sampling and Quantization, Pixels Relationship, Image Transforms: 2D FFT, Properties, Hadamard Transform, Discrete Cosine Transform, Haar Transform, Slant Transform, Hoteling Transform.

Image Enhancement: Image Enhancement in Spatial Domain, Enhancement Through Point Operation, Histogram Manipulation, Linear and Non-Linear Gray Level Transformation, Median Filter, Spatial Domain High-Pass Filtering, Filtering in Frequency Domain, Low Pass and High Pass Filter.

Image Restoration: Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration, Interactive Restoration. Morphological Image Processing: Dilation, Structuring Element Decomposition, Erosion, Combining Dilation and Erosion, Opening and Closing, The Hit or Miss Transformation.

Computer vision: Single camera models, feature detection and matching, object detection and recognition.

References

1. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2014.
2. S. Jayaraman, S. Esakkirajan, and T. Veerakumar, *Digital Image Processing*, TMH, 2010.
3. W. Burger and M. J. Burge, *Digital Image Processing: An Algorithmic Introduction*, 3rd ed. 2022.
4. S. E. Umbaugh, *Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIP Tools*, 2nd ed. CRC Press, 2011.
5. R. C. Gonzalez, R. E. Woods, and S. L. Eddings, *Digital Image Processing Using MATLAB*, 2nd ed. TMH, 2010.

CCE3121 Cloud and Edge Computing [3 1 0 4]

Introduction to Cloud Computing: Recent Trends in Computing Cloud Computing, Evolution of Cloud Computing. Cloud-enabling technologies.

Cloud Computing Architecture: Cloud deployment models, Service models in Cloud Computing, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS);

Virtualization: Virtualization technologies and architectures, internals of virtual machine monitors/hypervisors, virtualization of data centers; Case studies: Examples of Open Source and Commercial Clouds; Fog Computing; Role of Cloud in IoT, IoT Cloud.

Edge Computing: Introduction, Edge Computing Basics – Architectures, Applications, Edge Intelligence; Mobile Computing.

References:

1. R. Buyya, J. Broberg, and A. M. Goscinski, *Cloud Computing: Principles and Paradigms*, Wiley, 2011.
2. X. Zhou, H. Liu, and C. Shi, *Deep Learning on Edge Computing Devices: Design Challenges of Algorithm and Architecture*, Elsevier, 2022.
3. B. Sosinsky, *Cloud Computing Bible*, Wiley-India, 2010.
4. N. Antonopoulos and L. Gillam, *Cloud Computing: Principles, Systems and Applications*, Springer, 2012.
5. D. Marinescu, *Cloud Computing: Theory and Practice*, 3rd ed. Morgan Kaufmann, 2022.

CCE3140 Soft Computing [3 0 0 3]

Introduction: Soft computing and its applications; Neural networks: Architectures, Transfer Functions; Learning models: supervised, unsupervised, reinforcement learning.

Neural Network: Perceptron, backpropagation, multi-layer perceptron, radial basis function, recurrent neural network, self-organizing maps, Boltzmann machine.

Fuzzy Logic and Fuzzy Systems: Introduction and applications, fuzzy versus crisp set, basic operations on fuzzy sets, relations, fuzzy rule-based models, fuzzy classification, fuzzy arithmetic, fuzzy numbers, linguistic variables, arithmetic operations on intervals and numbers, lattice of fuzzy numbers, fuzzy equations, properties of membership functions, fuzzification and defuzzification, automated methods for fuzzy systems.

Genetic Algorithms: overview, applications, operators, fitness function, classifier systems, convergence; Hybrid soft computing approaches.

References

1. S. N. Sivanandam and S. N. Deepa, *Principles of Soft Computing*, 3rd ed. Wiley, 2018.
2. T. J. Ross, *Fuzzy Logic with Engineering Applications*, 2nd ed. Wiley, 2016.
3. S. J. Russell and P. Norvig, *Artificial Intelligence*, 4th ed. Pearson, 2022.
4. J. S. Jang, R. C. Sun, and E. Mizutani, *Neuro-fuzzy and Soft Computing*, Pearson, 2015

CCE3141 Information System Security [3 0 0 3]

Introduction: Information system security- Definition, scope, types & challenges. Security goals, security attacks, security services & mechanisms, security frameworks & standards- Overview & global security frameworks.

Cryptosystems: Symmetric key cryptography- Traditional & modern ciphers. Asymmetric key cryptography- RSA & Elgamal cryptosystem. Key management- Diffie Hellman key exchange. Hash function- Introduction, properties, algorithms, applications. Digital signatures- Working & applications. Digital certificates- Types, lifecycle & applications.

Network Security: Threats in networks, intruder detection & prevention. Firewalls- Types, policies & challenges. Network security protocols. Web security- Threats & tools. Secure Emails- Security mechanism, best practices.

Database Security: Database access control- Models, authorization mechanisms, audit & monitoring. Encryption for data at rest and in transit- Methods. Backup and recovery strategies- Types of backups, backup strategies, recovery strategies, challenges & best practices.

References:

1. W. Stallings, *Cryptography and Network Security: Principles and Practices*, 7th ed. Pearson Education, 2022.
2. B. A. Forouzan, *Cryptography and Network Security*, 3rd ed. McGraw Hill, 2015.
3. M. Whitman and H. Mattord, *Principles of Information Security*, 7th ed. 2022.
4. B. Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Wiley, ISBN 0-471-11709-9

CCE3142 Information System Auditing, Control, and Assurance [3 0 0 3]

Introduction to Information System Auditing: Definition and Purpose of IS Auditing, IS Audit Life Cycle. Auditing Standards: COBIT, ISO 27001, NIST Frameworks. Types of IS Audits: Compliance, Financial, Operational, and Integrated Audits.

Risk Management and IT Governance: Understanding IT Risks: Types and Sources, Risk Assessment and Management Processes, IT Governance Principles and Practices, Role of Auditors in Risk Mitigation and Governance.

Internal Controls in Information Systems: Concept and Objectives of Internal Controls, Control Frameworks: COSO and COBIT, General vs. Application Controls, Examples of Common IT Controls (Access Control, Change Management, etc.)

Audit Process and Techniques: Audit Planning and Scoping, Evidence Collection Techniques: Observation, Interviews, Sampling, and Data Analysis, Tools and Techniques for IS Auditing, Audit Reporting and Follow-up Actions.

Security Assurance and Emerging Trends: Concepts of Security Assurance and its Importance, Auditing Emerging Technologies: AI, IoT, Blockchain, and Cloud Security, Challenges in Auditing Modern IT Environments, Ethical and Legal Aspects in IS Auditing.

Case Studies and Practical Applications: Analysis of Real-World IS Audit Cases, Conducting a Mini-Audit on a Sample System.

References

1. J. J. Champlain, *Auditing Information Systems*, Wiley, 2003.
2. R. Weber, *Information Systems Control and Audit*, Pearson Education, 2007.
3. R. R. Moeller, *IT Audit, Control, and Security*, Wiley, 2010.
4. S. Senft and F. Gallegos, *Information Technology Control and Audit*, CRC Press, 2012.
5. K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, Pearson, 2019.

CCE3143 ADVANCED INTERNET TECHNOLOGIES [3 0 0 3]

Introduction: Need for web, web design fundamentals, website strategy and planning, tools for web development and testing, Modern Web browsers, Web Design Patterns – MVC, MVVM, Components, RESTful and GraphQL.

Web Servers: Architecture of Web Servers (IIS, Apache), Installation and configuration of Web Servers, Deployment of Web Pages, Maintenance and monitoring of Web pages, Security of Web Servers.

Data Exchange on the Web: JSON, XML, YAML, Text Formats, Binary formats, Data serialization and deserialization techniques, Real-Time Data Exchange and Streaming.

Advanced Client-Side Technologies: Advancements in HTML Living Standard, Responsive design with CSS preprocessors and Frameworks, Advanced JavaScript – ECMAScript 2015+

ReactJS: Introduction, React Components, React internals, Component intercommunication, Component styling and Performance optimizations. Advanced server-side technologies: Django – views, HTTP Methods, URL Routing, ORM, Templates, Forms Validation, Authentication and Authorization

References

1. B. Frain, *Responsive Web Design With HTML5 and CSS3*, 4th ed. Packt Publishing, 2022.
2. A. Osmani, *Learning JavaScript Design Patterns*, 2nd ed. O'Reilly Media, Inc., 2023.
3. E. Porcello and A. Banks, *Learning GraphQL: Declarative Data Fetching for Modern Web Apps*, O'Reilly Media, Inc., 2021.
4. S. Stefanov, *React: Up & Running*, 2nd ed. O'Reilly Media, Inc., 2020.
5. W. S. Vincent, *Django for Beginners: Build Websites with Python and Django*, 5th ed. Still River Press, 2022

CCE3144 Deep Learning [3 0 0 3]

Introduction: Perceptron, learning and recognition- supervise and unsupervised learning, Loss functions, backpropagation and stochastic gradient descent.

Convolutional Neural Networks: Introduction to Convnet, training a Convnet, weights initialization, batch normalization, pooling, padding, dropouts, hyperparameter optimization, CNN Architectures AlexNet, VGG, Inception, ResNet.

Recurrent neural network: Recurrent networks, long short-term memory(LSTM), gated recurrent units(GRU), recurrent neural network language models.

Deep unsupervised learning: Auto encoders, variation autoencoders, generative adversarial networks (GAN), maximum entropy distributions. Deep Belief Nets, Deep Boltzmann Machine, Convolutional Boltzmann Machine.

References

1. R. R. Russo, *Deep Learning for Beginners*, Kindle ed., 2019.
2. L. Deng and D. Yu, *Deep Learning: Methods and Applications*, 2nd ed. Now Publishers, 2018.
3. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. MIT Press, 2016.
4. M. Nielsen, *Neural Networks and Deep Learning*, 1st ed. Determination Press, 2015.
5. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

CCE314 Foundations of Blockchain Technology [3 0 0 3]

Introduction to Cryptographic Primitives: Encryption and decryption, public-key cryptography, ECC, digital signature, Secure Hash Algorithms, SHA-256, Keccak-256, Properties of Hash Algorithm.

Introduction to the Blockchain: The history of blockchain, Generic elements of a blockchain; Features of a blockchain; Applications of blockchain technology, Blockchain1.0, 2.0, and 3.0; Types of blockchain; Benefits and limitations of blockchain, Merkle Tree, Patricia Tree, Distributed Hash Table. Modelling faults and adversaries, Byzantine Generals problem, Blockchain – hash pointers, consensus, byzantine fault-tolerant distributed computing, Zero Knowledge proofs and protocols in Blockchain, Proof of Work (PoW), Proof of Stake (PoS) based Chains - Hybrid models (PoW + PoS)

Introducing Solidity: Solidity and Solidity files, Structure of a contract, State variables, Structure, Modifiers, Events, Enumeration, Functions, Data types in Solidity, Value types, passing by value and reference, Storage and memory data locations, Literals, Integers, Boolean, Byte data type, Arrays, Enumerations, Address, Mappings. Global Variables and Functions.

Expressions and Control Structures: Solidity expressions, decision conditional statements, control statements, the break statement, the continue statement, and the return statement; Smart contracts: Writing a simple contract, creating contracts, Using new keyword and address of the contract, Constructors, Contract composition. Inheritance, Encapsulation, Polymorphism, Method overriding, Abstract contracts, Interfaces, Functions, Modifiers, and Fallbacks.

References

1. A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S. Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*, 1st ed. Princeton University Press, 2016.
2. I. Bashir, *Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained*, 2nd ed. Packt Publishing, 2018.
3. R. Modi, *Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain*, 1st ed. Packt Publishing Ltd., 2018.
4. J. Thompson, *Blockchain: The Blockchain for Beginners, Guide to Blockchain Technology and Blockchain Programming*, CreateSpace Independent Publishing Platform, 2017.
5. M. Grincalaitis, *Mastering Ethereum: Implement Advanced Blockchain Applications Using Ethereum-supported Tools, Services, and Protocols*, Packt Publishing, 2019.

CCE3130 Design and Analysis of Algorithms Lab [0 0 2 1]

Implement a doubly linked list & BST, GCD Techniques, Bubble sort, Selection sort, Linear search, String Matching, sorting algorithms, DFS, BFS, Topological sorting, AVL tree, 2 -3 tree, Horspool algorithm, Open hast table, Floyd's algorithm, Warshall's algorithm, Greedy Techniques, Dijkstra's algorithm, Backtracking.

References

1. E. Horowitz, S. Sahni, and S. Rajasekaran, *Fundamentals of Computer Algorithms*, 2nd ed. University Press, 2008.
2. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 4th ed. MIT Press, 2022.
3. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, 1st ed. Pearson Education, 1999.
4. J. Kleinberg and É. Tardos, *Algorithm Design*, eTextbook ed. Pearson, 2021.
5. R. Neapolitan, *Foundations of Algorithms*, 5th ed. Jones and Bartlett Publishers, Inc., 2014.

CCE3131 Artificial Intelligence and Machine Learning Lab [0 0 2 1]

Implementation and/or use of libraries for application of algorithms: KNN, decision tree, naïve Bayes, support vector machine, regression, random forests, logistic regression, cross-validation, principal component analysis, k-means; Performance evaluation metrics: classification accuracy, logarithmic loss, confusion matrix, area under the curve, F1 score, mean absolute error, mean squared error. Frameworks: Python

References

1. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson Education, 2020.
2. T. M. Mitchell, *Machine Learning*, 1st ed. McGraw Hill, 2017.
3. D. Simon, *Evolutionary Optimization Algorithms*, 1st ed. Wiley, 2013.
4. D. Khemani, *A First Course in Artificial Intelligence*, 1st ed. McGraw Hill, 2015.
5. O. Richard, E. D. Peter, D. Hart, and G. Stork, *Pattern Classification*, 2nd ed. John Wiley, 2002.

VI Semester Syllabus

CCE3201 Next Generation Communication Systems [3 1 0 4]

Introduction to 1G/2G/3G/4G/5G terminology; evolution of public mobile services; Motivation for IP-based wireless networks: requirements and targets for long-term evolution (LTE); Technologies for LTE- 4G advanced features and roadmap evolutions from LTE to LTEA - wireless standards.

Review of Cellular Technologies: Modeling requirements and scenarios; Channel model requirements and Measurements; Propagation scenarios; METIS channel models; Map-based model; stochastic model; Comparison of Models.

Wireless Next-Generation Technologies: Next-generation networks; GSM technology; Introduction to next-generation networks (NGN); Broadband wireline and wireless alternatives; Wireless access technologies; Overview of TCP/IP and packet core; Advanced IP networking. Overview of voice and video transport over IP; NGN requirements; Architecture and protocols; Next-generation network and service management; NGN architectural components; NGN standards and protocols; NGN applications and architecture; SATCOM and broadband wireless architecture; NGN operations and management.

5G Network: Understand 5GPP & NGMN; 5G architecture and design objective; ITU-R IMT-2020 vision for 5G; 5G spectrum requirements; 5G RAN & dynamic CRAN; 5G NR logical architecture; 5G mobile edge computing & fog computing; millimetre wave propagation; Distributed massive MIMO principle; 5G ultra-dense networks; 5G CoMP; 5G air interface; 5G protocol stack; Machine Type Communication (MTC), Device to Device Communication (D2D), 5G Narrowband IoT.

References

1. N. Wilkinson, *Next Generation Networks Services, Technologies and Strategies*, 1st ed. Wiley, 2002.
2. R. Wood, *Next Generation Network Services*, Pearson Education, 2019.
3. S. Misra, *Wireless Communication and Networks: 3G and Beyond*, 2nd ed. McGraw Hill, 2017.
4. K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks*, Pearson Education, 2020.
5. D. Dulaimi, X. Wang, and C. Lin, *5G Networks: Fundamental Requirements, Enabling Technologies, and Operations Management*, 1st ed. John Wiley & Sons, 2018.

CCE3230 Next Generation Communication Systems Lab [0 0 2 1]

Fundamentals of SDR and wireless communication systems, Hands-on experience with SDR platforms, including the ADALM-PLUTO and USRP, develop skills in designing, implementing, and testing wireless communication systems using SDR platforms, Explore the latest advancements in SDR and wireless communication technologies.

References

1. N. Wilkinson, *Next Generation Networks Services, Technologies and Strategies*, 1st ed. Wiley, 2002.
2. R. Wood, *Next Generation Network Services*, Pearson Education, 2019.
3. S. Misra, *Wireless Communication and Networks: 3G and Beyond*, 2nd ed. McGraw Hill, 2017.
4. K. Pahlavan and P. Krishnamurthy, *Principles of Wireless Networks*, Pearson Education, 2020.
5. D. Dulaimi, X. Wang, and C. Lin, *5G Networks: Fundamental Requirements, Enabling Technologies, and Operations Management*, 1st ed. John Wiley & Sons, 2018.

CCE3240 Principles of Web Services [3 0 0 3]

Evolution and Emergence of Web Services: Emergence of web services and Service Oriented Architecture (SOA), introduction to web services –model of web services, tools and technologies enabling web services, benefits and challenges of using web services;

Web Service Architecture: characteristics, web services communication, WSDL, a brief overview of XML; SOA design implementation, managing SOA environment: service-oriented design process, design activities, determines services and tasks based on the business process model, implementing SOA.

SOAP (Simple Object Access Protocol): SOAP as a messaging protocol, UDDI architecture and implementation, UDDI with WSDL, UDDI specification.

REST (Representational State Transfer): messages, HTTP request and format, HTTP response and format, query parameters, protocol semantics of HTTP(GET, PUT, POST, DELETE, HEAD, OPTIONS, TRACE), REST vs SOAP.

References

1. R. Skoczylas and R. P. Sriganesh, *Developing Java Web Services*, 2nd ed. Wiley India, 2008.
2. S. Chatterjee and J. Webber, *Developing Enterprise Web Services*, Pearson Education, 2020.
3. F. Coyle, *XML, Web Services, and the Data Revolution*, 2nd ed. Addison-Wesley, 2018.
4. S. Graham, *Building Web Services with Java*, 3rd ed. Pearson, 2008.
5. B. M. Balachandar, *RESTful Java Web Services*, 3rd ed. Packt Publishing Limited, 2017.

CCE3241 Big Data Analytics [3 0 0 3]

Data Definitions and Analysis Techniques: Elements, Variables, and Data categorization, Levels of Measurement, Data management and indexing.

Data Analysis Techniques: Regression analysis, Classification techniques, Clustering, Association rules analysis. Unsupervised Learning, Recommendation Systems. Streaming Algorithms.

Big Data: Characteristics of Big Data, Data in a warehouse and Hadoop, Importance of Big Data, Big data use cases.

Hadoop: Distributed Architecture, HDFS, MapReduce, Spark, Similarity Search, Link Analysis.

NOSQL: NOSQL Models, Understanding Storage Architecture, Performing CURD operations, Querying NOSQL Stores.

Case Studies and Projects: Understanding business scenarios, Feature engineering and visualization, and Sensitivity Analysis.

References

1. R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye, *Probability and Statistics for Engineers and Scientists*, 9th ed. Pearson Education, 2021.
2. G. James, D. Witten, T. Hastie, and R. Tibshirani, *Statistical Learning: An Introduction to Statistical Learning*, Springer Texts in Statistics, vol. 103, Springer, New York, 2019.
3. H. Trevor, T. Robert, and F. Jerome, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. Springer-Verlag, New York, 2009.
4. J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets*, 3rd ed. Cambridge University Press, 2018.

CCE3242 Foundation of Digital Forensics [3 0 0 3]

Introduction to Digital Forensics: Definition, Need, and Scope of Digital Forensics, Overview of Digital Crime and Investigations, Types of Digital Evidence, Digital Forensics Process: Identification, Preservation, Analysis, and Presentation, Legal Framework and Ethics in Digital Forensics.

Digital Evidence and Acquisition: Sources of Digital Evidence: Computers, Networks, Mobile Devices, and Cloud Systems, Data Acquisition Methods: Static and Live Data Acquisition, Tools for Data Collection and Imaging: FTK Imager, EnCase, Chain of Custody and Documentation, Challenges in Evidence Collection.

File Systems and Data Recovery: Understanding File Systems: FAT, NTFS, EXT, and HFS+, Metadata Analysis and File System Forensics, Data Recovery Techniques: Deleted Files, Formatted Drives, and RAID Systems, Volatile Data Analysis: Memory and Cache.

Network and Mobile Forensics: Basics of Network Forensics: Packet Analysis, Intrusion Detection, and Traffic Logs, Tools for Network Analysis: Wireshark, Snort, Mobile Device Forensics: Acquisition and Analysis of Data from iOS and Android Devices, Challenges in Mobile Forensics.

Advanced Topics in Digital Forensics: Forensics in Virtualized and Cloud Environments, Anti-Forensics Techniques and Countermeasures, Malware Analysis and Reverse Engineering, Reporting and Presenting Digital Evidence in Court.

Case Studies and Practical Applications: Real-Life Case Studies of Cybercrime Investigations, Hands-on Exercises with Forensic Tools, Team-Based Mini-Project: Simulated Forensic Investigation.

References

1. E. Casey, *Digital Evidence and Computer Crime*, 3rd ed. Elsevier, 2020.
2. A. Nelson, P. Phillips, and S. Steuart, *Guide to Computer Forensics and Investigations*, 6th ed. Cengage Learning, 2020.
3. B. Carrier, *File System Forensic Analysis*, Addison-Wesley, 2005.
4. J. Sammons, *The Basics of Digital Forensics*, 2nd ed. Syngress, 2015.
5. H. Carvey, *Windows Forensic Analysis Toolkit*, 4th ed. Elsevier, 2014.

CCE4143 DevOps Fundamentals [3 0 0 3]

Introduction: Overview of DevOps, market trends, skills, Agile SDLC, a n d Concepts of TDD, BDD, delivery pipeline, and ecosystem.

Continuous Integration (CI): Version Control using Git, standard commands, working with remote repositories, branching and merging in Git, workflows, Git cheat sheet, introduction to Jenkins, Jenkins management, adding a slave node to Jenkins, building delivery pipeline, pipeline as a code, CI Strategies.

Continuous Testing (CT): Unit testing, Gherkin for automation testing, Selenium and Webdriver, creating test cases, handling different controls on webpages, and testing frameworks.

Docker ecosystem & Kubernetes: Introduction to container life cycle, Evolution of container orchestration, working with docker, publishing image, compose, Swarm, managing and running containers, Docker networking, network types, Kubernetes architecture and components, traditional vs. Kubernetes deployment strategies.

Continuous Deployment (CD): Different deployment strategies, Puppet as a tool for CD, installation and configuration, master and agent setup, puppet module, node classification, puppet environment and classes, automation and reporting.

Continuous Monitoring: Infrastructure Monitoring using Prometheus, Application Performance Monitoring (APM) using Datadog, H/W and OS level monitoring using Node Exporter, Network monitoring using Zabbix, Alert handling using Alertmanager, Visualization and dashboard using Grafana, Container or VM monitoring using cAdvisor, Logging using Loki.

DevOps on Cloud: Introduction to cloud computing, why DevOps on cloud, Introduction to AWS, various AWS services, DevOps using AWS.

References

1. G. Kim, J. Humble, P. Debois, J. Willis, and N. Forsgren, *The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations*, IT Revolution, 2021.
2. J. Davis and R. Daniels, *Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale*, O'Reilly Media, Inc., 2016.
3. T. Uphill, J. Arundel, N. Khare, H. C. C. Lee, H. Saito, and K. J. C. Hsu, *DevOps: Puppet, Docker, and Kubernetes*, Packt Publishing Ltd., 2017.
4. J. Arundel and J. Domingus, *Cloud Native DevOps with Kubernetes: Building, Deploying, and Scaling Modern Applications in the Cloud*, O'Reilly Media, 2019.
5. Y. Raheja, G. Borgese, and N. Felsen, *Effective DevOps with AWS: Implement Continuous Delivery and Integration in the AWS Environment*, Packt Publishing Ltd., 2018.

CCE3244 Recommender Systems [3 0 0 3]

Introduction: Introduction and basic taxonomy of recommender systems, traditional and non-personalized recommender systems, Applications of recommendation systems, and Issues with recommender systems.

Basics of Collaborative Filtering: latent factors, similarity measures, matrix factorization techniques for collaborative filtering, Regularizations, cold-start problem and their solutions.

Basics of Content-based Filtering: text-based approach, Feature normalization and preprocessing techniques, Problems and challenges with content-based approaches, Hybrid approaches of recommender systems, and different types of hybrid approaches.

Applications of Deep Learning in Recommender Systems: Neural collaborative filtering, MLP-based generalized matrix factorization, Autoencoders, RNN-based approaches, attention mechanism, graph neural network and reinforcement learning for recommender systems.

Evaluating Recommender System: Introduction, General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, Decision-Support metrics, and User-Centred metrics. Case study of recommender systems such as Netflix, Myntra, and Spotify.

Reference

1. D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*, Cambridge University Press, 2018.
2. F. Ricci, L. Rokach, and B. Shapira, *Recommender Systems Handbook*, 1st ed. Springer, 2011.
3. A. R. Kulkarni, A. Shivananda, A. Kulkarni, and V. A. Krishnan, *Applied Recommender Systems with Python*, Apress, 2023.
4. P. Singh, *Machine Learning with PySpark: With Natural Language Processing and Recommender Systems*, Apress, 2018.

CCE3245: Cloud Security [3 0 0 3]

Introduction to Cloud Computing: Recent Trends in Cloud Computing, Evolution of Cloud Computing. Cloud-enabling technologies.

Cloud Computing Architecture: Cloud deployment models, Service models in Cloud Computing, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service(SaaS);

Virtualization: Virtualization technologies and architectures, internals of virtual machine monitors/hypervisors, virtualization of datacenters;

Cloud Security Fundamentals: Identity and Access Management, Data Security in the Cloud, Challenges in security in Multi-cloud and Hybrid Environments; Fog Computing; Role of Cloud in IoT, IoT Cloud; Introduction to Edge Computing: Edge Computing Basics, Architectures, Applications, Edge Intelligence; Mobile Computing.

Case studies of Open Source and Commercial Clouds: OpenStack, CloudStack, AWS, Microsoft Azure, GCP.

References

1. G. Thompson, *CCSK Certificate of Cloud Security Knowledge All-in-One Exam Guide*, McGraw Hill, 2020.
2. R. Catlin, *Enterprise Cloud Security and Governance: Efficiently Set Data Protection and Privacy*, Packt Publishing, 2017.
3. *Cloud Security: Concepts, Applications and Perspectives*, CRC, 2021.
4. R. L. Krutz and R. D. Vines, *Cloud Security: A Comprehensive Guide to Secure Cloud Computing*, 2020.

CCE3246: Full Stack Development [3 0 0 3]

Introduction to Web Development: Comprehensive overview of web technologies (Front-end, Back-end, Databases), understanding the MERN stack, setting up the development environment, Software Development Lifecycle (SDLC) for web applications, and introduction to version control tools (Git, GitHub).

HTML, CSS & JavaScript: HTML Elements, Forms, and Semantic Tags, CSS Styling, Flexbox, and Grid Layouts, Responsive Web Design (Media Queries). CSS Preprocessors (SASS/SCSS). JavaScript Fundamentals: Data Types, Functions, and DOM Manipulation, ES6 Features (let, const, arrow functions, template literals).

React JS - Introduction to React JS, JSX, Rendering Elements, Components and Props, State and Lifecycle, Handling Events, Conditional Rendering, Lists and Keys, Forms, Composition vs Inheritance, CSS in JS, Routing-Query parameters & Path parameters.

Node JS- Node JS Overview, Basics and Setup, Console, Command Utilities, Modules, Events, Node JS with Express JS, Node JS Database Access.

Database: SQL and NoSql Concepts, Create and Manage MongoDB, MongoDB with NodeJS, Services Offered by MongoDB. **Projects:** Full stack web application.

References

1. J. Duckett, *Web Design with HTML, CSS, JavaScript, and JQuery Set*, Professional JavaScript for Web Developers, N. C. Zakas, 2012.
2. R. Nixon, *Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites*, 2019.
3. A. Mardan, *Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB*, 2015.
4. J. J. Patrick, *SQL Fundamentals*, 2008.
5. J. Fawcett, D. Ayers, and L. R. E. Quin, *Beginning XML*, 2019.

CCE3247 Image and Video Analytics [3 0 0 3]

Basics of Image Processing: Image representation: pixels, color spaces, and image formats; Common image processing operations: smoothing, sharpening, and morphological operations. Introduction to Video Analytics: Temporal data and its significance in videos. Video frame rates and resolutions. Real-world applications of image and video analytics, Key Concepts in Analytics

Image Features: Color, texture, and shape. Edge detection: Gradient-based methods (Sobel, Prewitt, Canny). Keypoint detection and descriptors: SIFT, SURF, ORB. Video Features: motion vectors, optical flow. Frame-based versus sequence-based features. Feature selection and dimensionality reduction techniques (e.g., PCA, LDA).

Object Detection: Region proposals, YOLO, and SSD. Semantic segmentation: pixel-level image understanding, Face recognition, Video Understanding: Action and activity recognition: temporal modeling techniques. Video summarization and keyframe extraction. Event detection and anomaly detection in videos.

Deep Learning in Image and Video Analytics: Convolutional Neural Networks (CNNs) for image analysis, Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) for video analytics. Transformer-based models for vision tasks. Multi-Modal Analytics: Combining video with audio and textual information. Applications of multi-modal approaches (e.g., video captioning, sentiment analysis). Privacy concerns in video surveillance.

Applications and Emerging Trends: Healthcare, Security, Autonomous Systems, Entertainment, Edge computing for real-time analytics, Explainable AI in image and video analysis, Future of 3D image and video processing.

References

1. R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2018.
2. R. Szeliski, *Computer Vision: Algorithms and Applications*, 2022.
3. Y. Wang, J. Ostermann, and Y.-Q. Zhang, *Video Processing and Communications*, 2019.
4. M. Elgendy, *Deep Learning for Vision Systems*, 2015.
5. R. Chellappa and A. K. Roy-Chowdhury, *Vision-Based Human Motion Analysis*, 2016.
6. M. Tavakoli, H. R. Rabiee, and A. Shahroudy, *Action Recognition in Videos: A Deep Learning Perspective*, 2020.

CCE3248 Network Vulnerabilities [3 0 0 3]

Network Vulnerabilities: Types of Vulnerabilities: Configuration, Protocol, Software, and Hardware Vulnerabilities, Vulnerability Assessment and Management: Tools and Techniques, Network Scanning and Mapping: Nmap, Nessus, Exploiting Vulnerabilities: Common Methods and Tools, Case Studies of Real-World Network Vulnerabilities.

Cryptography and Network Protection: Basics of Cryptographic Systems: Symmetric and Asymmetric Encryption, Secure Communication Protocols: SSL/TLS, IPsec, VPNs, Public Key Infrastructure (PKI) and Certificate Management, Application of Cryptography in Network Security, Common Cryptographic Vulnerabilities.

Wireless and Mobile Network Vulnerabilities: Security Issues in Wireless Networks (Wi-Fi): WEP, WPA, WPA2, and WPA3, Attacks on Wireless Networks: Eavesdropping, Rogue Access Points, and Evil Twin Attacks, Mobile Network Security: Cellular and IoT Vulnerabilities, Security Practices for Wireless and Mobile Networks.

Advanced Topics in Network Vulnerabilities: Zero-Day Exploits and Advanced Persistent Threats (APTs), Network Forensics: Capturing and Analyzing Traffic, Security Challenges in Cloud and Virtualized Environments, Emerging Threats and Trends: Quantum Computing, AI-Driven Attacks.

Case Studies and Practical Applications: Analysis of Historical Network Security Breaches, Hands-On Exercises with Vulnerability Scanning Tools, Mini-Project: Designing and Securing a Simulated Network.

References

1. W. Stallings, *Network Security Essentials: Applications and Standards*, 6th ed., Pearson Education, 2020.
2. C. Kaufman, R. Perlman, and M. Speciner, *Network Security: Private Communication in a Public World*, Pearson Education, 2021.
3. B. A. Forouzan, *Cryptography and Network Security*, 3rd ed., McGraw-Hill, 2015.
4. K. Mitnick and W. L. Simon, *The Art of Intrusion*, Wiley, 2015.
5. M. T. Simpson, K. Backman, and J. E. Corley, *Hands-On Ethical Hacking and Network Defense*, 4th ed., Cengage Learning, 2017.

CCE3249 Microservices Architecture [3 0 0 3]

Introduction: Overview of web technologies, monolithic vs microservices architecture, principles of scalability and resilience, Software Development Lifecycle for microservices, introduction to version control tools. **Node.js, Express.js & Databases:** Node.js overview, basics and setup, console, command utilities, modules, events, Express.js fundamentals, REST API development with Express.js, SQL and NoSQL concepts, create and manage MongoDB, MongoDB with Node.js, services offered by MongoDB, polyglot persistence with SQL and NoSQL. **Service Communication & Event-Driven Microservices**: RESTful API design and documentation, API gateway concepts and implementation, inter-service communication patterns, error handling, idempotency, rate limiting, message brokers, eventual consistency, saga pattern for distributed transactions, notifications, and background jobs. Resilience, **Security & Performance:** Resilience patterns, Redis for caching and session management, load balancing and fault tolerance, performance optimization in Node.js services, authentication and authorization, JWT, OAuth2/OIDC, service-to-service authentication, secrets management and environment variables. **Containerization, Orchestration, Observability & CI/CD Docker:** containerizing Node.js microservices, Kubernetes deployment, centralized logging, unit testing, integration testing, contract testing, load and stress testing, Actions for CI/CD pipelines

References:

1. Microservice Architecture - Aligning Principles, Practices, and Culture (First Edition) by Irakli Nadareishvili, Ronnie Mitra, Matt McLarty & Mike Amundsen, Oreilly (2016)
2. Essentials of Microservices Architecture: Paradigms, Applications, and Techniques - Chellammal Surianarayanan, Gopinath Ganapathy, Raj Pethuru, CRC Press (2021)
3. Building Microservices - Designed Fine-Grained Systems (First Edition) by Sam Newman, Orielly (2015)
4. Distributed Systems with Node.js: Building Enterprise-Ready Backend Services - Thomas Hunter II, Orielly (2020).

VII Semester Syllabus

CCE4140 Social Network Analysis [3 0 0 3]

Introduction to Social Web: Nodes, Edges and Network measures, Describing Nodes and Edges, Describing Networks, Layouts; **Visualizing Network features:** The role of Tie Strength, Measuring Tie Strength, Tie Strength and Network Structure, Tie Strength and Network Propagation, Link Prediction, Entity Resolution; **Link Prediction:** Case Study Friend Recommendation, Introduction to Community Discovery, Communities in Context, Quality Functions; **Algorithms:** The Kernighan-Lin algorithm, Agglomerative Algorithms, Spectral Algorithms, Multi-level Graph Partitioning, Markov Clustering, Other Approaches; **Introduction to Social Influence**: Influence Related Statistics, Social Similarity and Influence, Homophile, Existential Test for Social Influence, Influence and Actions, Influence and Interaction, Influence Maximization in Viral Marketing, Tools: Tulip, State Net, and Social Network Analyzer.

References:

1. J. Goldbeck, "Analysing the Social Web", Morgan Kaufmann Publications, 2013.
2. C. C. Aggarwal, "Social Network Data Analytics", Springer Publications, 2011.
3. J. Scott, "Social Network Analysis", (4e), SAGE Publications Limited, 2019.
4. John Scott, Social Network Analysis: Research Methods ,2024.
5. S. Kumar, F. Morstatter, H. Liu, "Twitter Data Analytics", Springer Publications, 2013

Course Outcomes:

CO1: Describe fundamental concepts of social web, including nodes, edges, network measures, and visualization techniques. (Bloom's Level 2)

CO2: Explain network features such as tie strength, link prediction, and community discovery using appropriate algorithms. (Bloom's Level 2)

CO3: Apply graph partitioning, clustering, and community detection algorithms to analyze social networks and recommend connections. (Bloom's Level 3)

CO4: Analyse social influence, similarity, and viral marketing strategies to evaluate their impact on network interactions and information propagation. (Bloom's Level 4)

CCE4141: Human-Computer Interaction [3 0 0 3]

Introduction to Interaction Design: User Experience, Process of Interaction Design, Understanding and Conceptualizing Interaction, Conceptual Models, Interface Metaphors, Interaction Types, Cognitive Aspects, Cognition, Cognitive Framework. **Interfaces:** Types of Interfaces, Natural User Interfaces, Data Gathering and Key Issues, Data Recording, Interviews, Questionnaires, Observation, Choosing and Combining Techniques, Data Analysis, Interpretation and Presentation of Qualitative and Quantitative Analysis, Simple Analysis, Tools and Theoretical Frameworks, Presenting the Findings. **Process of Interaction Design:** Establishing Requirements, Data Gathering for Requirements, Task Description, Task Analysis, Design, Prototyping and Construction, Conceptual Design and Physical Design, Using Scenarios, Prototypes in Design. **Design Process:** Interaction Design Basics and Rules, Software Lifecycle, Universal Design and Multimodal Interaction, Design for Diversity, GOMS, Design tools: Figma, Unity.

References:

1. H. Sharp, Y. Rogers, and J. Preece, "Interaction Design: Beyond Human-Computer Interaction", 6th Edition, John Wiley & Sons, 2023.
2. Dix, J. E. Finlay, G. D. Abowd, and R. Beale, "Human-Computer Interaction", 3rd Edition, Pearson Education, 2004.
3. W. O. Galitz, "The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques", 3rd Edition, John Wiley & Sons, 2007.
4. D. Benyon, "Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design", 4th Edition, Pearson Education, 2019.
5.

Course Outcomes:

CO1: Identify the key principles for designing user-centred and highly usable software systems. (Bloom's Level 1)

CO2:Examine the development of Human-Computer Interaction (HCI) theories and practices. (Bloom's Level 4)

CO3: Adapt various methods, quality metrics, and data analysis techniques related to HCI. (Bloom's Level 3)

CO4: Assess interaction challenges from technical, cognitive, and functional viewpoints to enhance entrepreneurial skill development. (Bloom's Level 5)

CCE4142: Natural Language Processing [3 0 0 3]

Introduction to NLP & Language Fundamentals: History, evolution, and phases of NLP, Types of Ambiguity in Language, Basic Text Processing Techniques, Text Classification and Sentiment Analysis, Evaluation Metrics, Applications: Spam Filtering, Sentiment Analysis. **Classical Language Models:** Word, Sentence, Document Representation, N-gram Models and Smoothing Techniques, Perplexity and Evaluation Metrics, Hidden Markov Models, and POS Tagging. Semantic Analysis and Word Embeddings: Lexical Semantics and WordNet, Word Sense Disambiguation, Word Embeddings: Word2Vec, GloVe, FastText, Speech Recognition Techniques, **Neural NLP and Sequence Models:** Introduction to Neural Networks for NLP, RNNs, GRUs, LSTMs – Architecture and Applications, Named Entity Recognition, Sequence Labeling, Encoder-Decoder Models for NLP Tasks. **Transformer Models and Pretrained LMs:** Self-Attention and Transformer Architecture, Pretrained Models Overview: BERT, GPT, RoBERTa, Transfer Learning and Fine-Tuning, Decoding Techniques: Beam Search, Sampling Strategies. **NLP Tools, Ethics and Trends:** Overview of Corpora: Treebank, Wikipedia, SentiWordNet, Tools & Libraries: NLTK, spaCy, HuggingFace Transformers, Bias, Fairness, and Ethical Considerations in NLP, Latest Trends: Multilingual NLP, Prompt Engineering, LLMs.

References:

1. Daniel Jurafsky and James H. Martin, 'Speech and Language Processing', 3rd Edition (Draft), Pearson.
2. Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze, 'Introduction to Information Retrieval', Cambridge University Press.
3. Steven Bird, Ewan Klein, and Edward Loper, 'Natural Language Processing with Python', O'Reilly Media.
4. Yoav Goldberg, 'Neural Network Methods in Natural Language Processing', Morgan & Claypool Publishers.
5. Delip Rao and Brian McMahan, 'Natural Language Processing with PyTorch', O'Reilly Media.

Course Outcomes:

CO1: Understand the linguistic and computational foundations of NLP. (Bloom's Level 2)
CO2: Apply classical NLP techniques such as tokenization, stemming, and POS tagging. (Bloom's Level 3)
CO3: Analyze and implement statistical models and text classification techniques. (Bloom's Level 4)
CO4: Develop NLP applications using neural networks, word embeddings, and transformer models. (Bloom's Level 6)
CO5: Evaluate NLP systems using standard metrics and understand ethical and recent trends. (Bloom's Level 5)

CCE4143: Mobile Application Development **[3 0 0 3]**

Android and its Tools: Introduction, open handset alliance, ecosystem overview, need and features, tools and software required for application development, system architecture. **Application Design Essentials:** Anatomy of an application, key terminologies, application context, activities, services, intents, receiving and broadcasting intents, development tools, manifest file and common settings, using intent filters, permissions, steps to install and configure android studio and sdk. **User Interface Design:** User interface screen elements, designing interfaces with layouts, and working with animations. **Testing, Security and Applications Deployment:** Testing techniques, publishing process, using preferences, managing resources in a hierarchy, and working with different resource types. Android Security Models. Applications Deployment: creating, signing, and deploying on different platforms: **Common APIs:** Data and storage access, managing data with SQLite, sharing data using content providers, networking, web services, telephony, and deploying applications.

References:

1. Neil Smyth, Android Studio Koala Essentials – Java Edition, Payload Media, 2024
2. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2011.
3. Reto Meier, "Professional Android 2 Application Development", Wiley India Pvt Ltd, 2010.
4. Mark L Murphy, "Beginning Android", Wiley India Pvt Ltd, 2012.
5. Android Application Development All in one for Dummies by Barry Burd, Edition: I, 2015

Course Outcomes :

CO1: Describe fundamental concepts of social web, including nodes, edges, network measures, and visualization techniques. (Bloom's Level 2)

CO2: Explain network features such as tie strength, link prediction, and community discovery using appropriate algorithms. (Bloom's Level 2)

CO3: Apply graph partitioning, clustering, and community detection algorithms to analyze social networks and recommend connections. (Bloom's Level 3)

CO4: Analyse social influence, similarity, and viral marketing strategies to evaluate their impact on network interactions and information propagation. (Bloom's Level 4)

CCE4144: Information Retrieval [3 0 0 3]

Basic concepts of IR: data retrieval and information retrieval, IR system block diagram, automatic text analysis, Luhn's ideas, conflation algorithms, indexing and index term weighing, probabilistic indexing, automatic classification, similarity scoring, measures of association, different matching coefficients, and feedback analysis. Scoring, term weighting & the vector space model. **Machine learning in IR:** Automatic classification & clustering in IR, clustering algorithms, dendrograms, Learning to Rank, diversity, fairness, bias. **Classification methods:** cluster hypothesis, clustering algorithms, single-pass algorithm, single link algorithm, Rochhio's algorithm, and Word embedding techniques: Word2Vec, GloVe, FastText. **Parallel and Distributed IR:** Parallel SIMD and MIMD architectures, distributed IR – collection partitioning, source selection, query processing. **File structures and searching algorithms:** inverted file, suffix trees and suffix arrays, signature files, ring structure, search strategies, Boolean search, serial search, and cluster-based retrieval. **IR models:** basic concepts, Boolean model, vector model; probabilistic model, Set-Theoretic Models: Extended Boolean models, Fuzzy set model, models for browsing. Trends and research issues.

References:

1. C.D. Manning, P. Raghavan, H. Schuetze, Introduction to Information Retrieval, (1e), Cambridge University Press, 2007.

2. B.Croft, D.Metzler, T. Strohman Search Engines: Information Retrieval in Practice, (2e), Pearson Education, 2015.

3. B. Ricardo, B.Neto Modern Information Retrieval, (2e), Addison-Wesley, 2011.

4. Garg, M. Natural Language Processing and Information Retrieval: Principles and Applications. CRC Press, 2023.

Course Outcomes:

CO1. Explain the fundamental concepts, architecture, and classical models of Information Retrieval systems. (Bloom’s Level 2)

CO2. Apply indexing techniques, similarity measures, and term-weighting to implement retrieval tasks. (Bloom’s Level 3)

CO3. Analyze the role of machine learning in IR by applying clustering, classification, and Learning to Rank approaches, while considering fairness and bias. (Bloom’s Level 4)

CO4. Evaluate the efficiency of file structures, distributed IR techniques, and parallel architectures for large-scale information retrieval. (Bloom’s Level 5)

CO5. Assess current research issues and trends in IR and propose potential solutions with ethical and practical considerations. (Bloom’s Level 5)

CCE4145 Wireless Sensors & Ad hoc Networks [3 0 0 3]

Introduction to Wireless Sensor & Ad Hoc Networks: Characteristics and challenges Applications of WSN, Characteristics of ad hoc networks, Differences between WSN, Ad hoc, and traditional networks, **Medium Access Control Protocols**: Fundamentals of MAC in wireless networks, Contention-based protocols: IEEE 802.11, S-MAC, T-MAC, Schedule-based protocols, Energy-efficient MAC design issues, QoS and fairness considerations **Routing Protocols in WSN & Ad Hoc Networks**: Routing challenges in dynamic, resource-constrained environments, Proactive vs. reactive vs. hybrid routing protocols: DSDV, AODV, DSR, ZRP, WSN-specific routing protocols: LEACH, PEGASIS, TEEN, SPIN, Directed Diffusion, Location-based and QoS-aware routing, Case studies **Transport, Data Management, and Security**: Transport layer issues in WSN, Congestion control & reliability in sensor networks, Data-centric storage and querying: TinyDB concepts, Time synchronization in WSN, Security challenges in WSN and MANETs, Key management & trust models; **Energy Management & Cross-layer Design**: Energy efficiency challenges in WSN, Power-aware communication techniques, Duty cycling & energy harvesting techniques, Cross-layer optimization strategies; **Applications & Emerging Trends**: WSN in IoT, Cyber-Physical Systems, Industry 4.0,Smart agriculture, healthcare monitoring and other emerging applications.

References:

1. Karl, H., & Willig, A. *Protocols and Architectures for Wireless Sensor Networks*. Wiley, 2005.
2. H. M. A. Fahmy. Wireless Sensor Networks: Energy Harvesting and Management Techniques. Springer, 2020.
3. Murthy, C. S. R., & Manoj, B. S. *Ad Hoc Wireless Networks: Architectures and Protocols*. Pearson Education, 2004.
4. Sohraby, K., Minoli, D., & Znati, T. *Wireless Sensor Networks: Technology, Protocols, and Applications*. Wiley-Interscience, 2007.
5. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. "Wireless Sensor Networks: A Survey." *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.
6. Wireless Ad-hoc and Sensor Networks." (Singh, P., et al.) Taylor & Francis, 2024.

Course Outcomes:

CO1: Explain the characteristics and architecture of wireless sensor and ad hoc networks.(Bloom's Level 2)

CO2: Compare different medium access control (MAC) and routing protocols for WSNs and MANETs. (Bloom's Level 4)

CO3: Analyze the design issues related to energy, scalability, and QoS in sensor networks. Bloom's Level 2

CO4: Apply WSN principles to real-life applications in IoT and pervasive computing. (Bloom's Level 3)

CO5: Demonstrate knowledge of security and reliability challenges in WSNs. (Bloom's Level 4)