DAYANANDA SAGAR
UNIVERSITY

PROGRAMME PROJECT REPORT (PPR)

FOR

BACHELOR OF COMPUTER APPLICATIONS (BCA)

Mode: ONLINE

CENTRE FOR DISTANCE AND ONLINE EDUCATION (CDOE)

DAYANANDA SAGAR UNIVERSITY

BENGALURU

Dayananda Sagar University Bangalore

28/8/25

Registrar

List of Contents

PROGRAMME PROJECT REPORT

Introduction

The digital age has transformed every element of life, making technology a crucial component of society, business, and education. In recognition, Dayananda Sagar University (DSU) has developed programs that equip learners to succeed in a modern, technologically advanced, and interconnected society. At DSU, we strongly emphasise giving learners the abilities and information needed to navigate the ever-changing digital landscape successfully.

Dayananda Sagar University's Centre for Distance and Online Education (CDOE) offers an online Bachelor of Computer Applications (BCA) program to give learners a solid foundation in IT solutions and computer applications. By combining academic understanding with real-world application, the program ensures that learners are prepared to tackle difficulties in the technology industry. It tackles current technological and societal concerns while promoting creativity and problem-solving. The curriculum guarantees learners realise their full potential through a committed and cooperative learning environment facilitated by knowledgeable professors.

Learners enrolled in the online BCA program gain the capacity to think critically, work effectively, and adjust to new technology. The program emphasises technical proficiency, logical reasoning, and good communication, enabling learners to impact technological decisions and significantly contribute to the IT industry. The curriculum develops these abilities and turns learners into tech-savvy experts who can propel innovation and change.

Three years of undergraduate study are required for the online Bachelor in Computer Applications (BCA) degree, which offers elective concentrations in software development, cybersecurity, data analytics, and artificial intelligence. These electives, designed to satisfy industry expectations, help learners acquire proficiency in cutting-edge technology. The program's multidisciplinary approach equips learners to succeed in various fields, including database administration, web development, software engineering, and IT consulting.

The Online BCA Program gives learners the abilities and competencies needed to succeed in the IT sector, which is becoming increasingly dependent on digital solutions. It equips them to think creatively and become proficient professionals who can handle technical difficulties and influence the direction of digital change in the future. Learners who select the program set a path to a bright and influential career in the rapidly changing technology industry.

1. Programme's Mission and Objectives

The mission of CDOE's Online BCA Program at Dayananda Sagar University (DSU) is to give learners the information and abilities they need to succeed in the quickly changing field of information technology. The curriculum combines technological know-how, ethical behaviour, and critical thinking to enable learners to lead innovation, tackle practical IT problems, and significantly contribute to society's digital transformation.

Mission

To impart technical knowledge through innovative teaching methodologies, cutting-edge research, and consultancy initiatives. The program's state-of-the-art facilities and guidance from internationally recognised faculty ensure learners receive a world-class education. The curriculum is regularly updated to meet the dynamic needs of the IT industry, fostering relevance and adaptability. The program bridges the gap between academia and real-world applications by promoting meaningful partnerships with industry and community stakeholders. It strives to produce competent graduates who embody professional ethics, critical life skills, and a commitment to career excellence.

Objectives

- To prepare graduates for successful careers as Computer Programmers, Algorithm Developers., and Application Developers.
- To enable graduates to pursue professional careers in industry, government, academia, and the military as innovative IT professionals.
- To encourage graduates to continue learning and advancing their careers through participation in professional organisations, attaining certifications, and pursuing higher education.

- To develop graduates as active members of society, ready to contribute and serve locally and internationally.

The online BCA program aims to give learners a thorough understanding of systems design, programming, and computer applications. Practicals, projects, and case studies strongly emphasise hands-on learning and guarantee that learners have real experience addressing challenging IT issues. The program's integration of cutting-edge technologies and industry-relevant techniques equips learners to navigate and contribute to local and global IT ecosystems successfully.

The outcomes from the programme focus on developing critical, analytical, and problem-solving skills, enabling graduates to apply computing and mathematical knowledge effectively. It prepares learners to identify and analyse complex computing problems, design and implement computational solutions, and evaluate their effectiveness. The programme emphasises proficiency in modern tools and techniques, fostering teamwork and leadership in multidisciplinary settings. Graduates are equipped to communicate effectively through written and oral presentations, collaborate seamlessly, and apply project management and software engineering practices to meet stakeholder needs. Furthermore, the programme instils the ability for lifelong learning and professional growth, ensuring graduates remain adaptable and innovative in a rapidly evolving technological landscape.

The Online BCA Program equips learners to succeed as IT professionals, innovators, and leaders in the digital age by fusing academic knowledge with real-world application.

2. Relevance of Programme with Dayananda Sagar University Mission and Goals

The Vision of Dayananda Sagar University emphasises becoming a leading institution excelling in education, research, innovation, and entrepreneurship. It aims to nurture individuals with exceptional leadership skills, empowering them to address national and global challenges, driving positive change, and contributing to societal development. The focus is on excellence and holistic development to shape future leaders.

The mission statement of Dayananda Sagar University emphasises fostering a supportive environment that encourages creativity, innovation, and academic excellence. By aligning all efforts with the institution's vision, the university aims to achieve its goals while nurturing

intellectual growth, inspiring new ideas, and maintaining a commitment to excellence in every pursuit.

Vision

To emerge as the powerhouse of information Technology and Allied areas, developing competent computer professionals to meet the dynamic needs of disruptive technologies.

Mission

We aim to achieve our objectives in an environment that enhances creativity, innovation, and scholarly pursuits while adhering to our vision.

The online BCA program fosters a dynamic learning environment that combines academic achievement, creativity, and practical training in computer applications, all of which contribute to Dayananda Sagar University's mission. The curriculum strongly emphasises producing entrepreneurs and IT specialists who are prepared to tackle domestic and international issues in a quickly changing technology environment. It embodies the university's goal by encouraging scholarly endeavours, creativity, and intellectual development. With a curriculum that fosters technical proficiency, problem-solving skills, and leadership, the Online BCA Program equips learners to succeed in the workplace and improve society and business.

3. Nature of Prospective Target Group of Learners

The Online BCA Programme from CDOE is designed for learners aiming to build a strong foundation in the IT and computing fields. This program is ideal for recent school graduates who aspire to secure entry-level roles in technology and individuals seeking to enhance their knowledge and skills in programming, database management, and software development. The curriculum equips learners with the essential tools to excel in the evolving IT industry.

The program's affordability, accessibility, and flexibility make it an excellent choice for learners from diverse backgrounds. It allows them to pursue their education without needing relocation or commuting. Delivered online, the program fosters inclusivity and collaboration, contributing to India's National Education Policy goal of achieving a 50% Gross Enrollment Ratio (GER) by 2035.

4. Appropriateness of programme to be conducted in Online mode to acquire specific skills and competence

To enable the learners of the online BCA programme to learn, the courses are delivered through Self-Learning e-Modules, offering a structured and engaging approach to e-learning. These modules are designed to be self-explanatory, providing clear instructions and content to facilitate independent learning. They are self-contained, encompassing all necessary resources within a single unit, and self-directed, allowing learners to progress at their own pace. The modules include self-evaluation components to assess learners' understanding and progress. This approach ensures learners can effectively acquire the prescribed knowledge and skills while enjoying flexibility, accessibility, and a personalised learning experience tailored to their needs. The course study material for the online programme is made available through the four-quadrant approach and will be uploaded to the University's Learning Management System (LMS). The components of the four quadrants are:

(i) e-Tutorial: in the form of recorded faculty lecture videos.

(ii) e-Content: Text Materials – e-SLM and access to the online library for additional reading materials.

(iii) Discussion forum: For raising doubts and clarifying the same by the Course Coordinators/Course Mentors assigned to learners

(iv) Assessments: Self-Assessments in the form of quizzes, fill in the blanks, matching questions, short answer questions, and internal assessments in the form of assignments to reinforce learning.

The online BCA programme from CDOE is thus designed to align perfectly with the online learning format, ensuring effective delivery and engagement. Its structure and content cater to the needs of learners learning online, hence making it a suitable choice for those seeking a flexible and accessible higher education opportunity.

5. Instructional Design

5.1 Programme Curriculum

Experts in computing and information technology have meticulously developed the Online BCA Program's curriculum, carefully combining courses on cutting-edge technologies and modern IT procedures with fundamental computer science ideas. The balanced approach ensures that learners acquire state-of-the-art technical abilities while encouraging a sense of accountability for

technology's moral and sustainable use. Learners who complete this curriculum will have the values and knowledge needed to succeed in the ever-changing IT industry. The syllabus satisfies current academic and industry criteria and is approved by the Board of Studies by university policies.

SEM	SN.	Course Code	TITLE OF THE COURSE	CREDITS
1	1	BCAOL101	MATHEMATICS-I	4
1	2	BCAOL102	FUNDAMENTALS OF PROGRAMMING	3
1	3	BCAOL103	COMPUTER ORGANIZATION	3
1	4	BCAOL104	WEB PROGRAMMING	4
1	5	BCAOL105	ACCOUNTING AND FINANCIAL MANAGEMENT	4
1	6	BCAOL106	ENGLISH IN PRACTICE	3
1	7	BCAOL107	CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS	1
1	8	BCAOL108	PROGRAMMING LAB	1
1	9	BCAOL109	WEB PROGRAMMING LAB	1
				24
2	10	BCAOL201	MATHEMATICS-II	4
2	11	BCAOL202	OBJECT-ORIENTED PROGRAMMING USING JAVA	3
2	12	BCAOL203	DATA STRUCTURES	3
2	13	BCAOL204	COMPUTER ARCHITECTURE	4
2	14	BCAOL205	TECHNICAL COMMUNICATIONS	3
2	15	BCAOL206	ENVIRONMENT AND PUBLIC HEALTH	1
2	16	BCAOL207	OOP USING JAVA LAB	1
2	17	BCAOL208	DATA STRUCTURES LAB	1
2	18	BCAOL209	ENGLISH COMMUNICATION LAB	1
2	19	BCAOL210	MINI PROJECT ON WEB PROGRAMMING	2
				23
3	20	BCAOL301	SOFTWARE ENGINEERING AND TESTING	4
3	21	BCAOL302	PYTHON PROGRAMMING	3
3	22	BCAOL303	MATHEMATICS-III	4
3	23	BCAOL304	COMPUTER NETWORKS	4
3	24	BCAOL305	ANALYSIS AND DESIGN OF ALGORITHMS	4
3	25	BCAOL306	OPERATING SYSTEMS	3
3	26	BCAOL307	PYTHON PROGRAMMING LAB	1

3	27	BCAOL308	OPERATING SYSTEMS LAB	1
3	28	BCAOL309	ANALYSIS AND DESIGN OF ALGORITHMS LAB	1
3	29	BCAOL310	TERM PAPER	1
				26
4	30	BCAOL401	OBJECT-ORIENTED ANALYSIS AND DESIGN	4
4	31	BCAOL402	WEB TECHNOLOGIES	3
4	32	BCAOL403	DATABASE SYSTEMS	3
4	33	BCAOL404	DATA SCIENCE WITH PYTHON	4
4	34	BCAOL405X	ELECTIVE - I	4
4	35	BCAOL406	LIBERAL STUDIES - I	2
4	36	BCAOL407	DATA SCIENCE WITH PYTHON LAB	1
4	37	BCAOL408	DATABASE SYSTEMS LAB	1
4	38	BCAOL409	MINI PROJECT	2
				24

5	39	BCAOL501	INFORMATION AND NETWORK SECURITY	3
5	40	BCAOL502X	ELECTIVE - II	4
5	41	BCAOL503X	ELECTIVE - III	4
5	42	BCAOL504	WEB TECHNOLOGIES LAB	1
5	43	BCAOL505	PROJECT STAGE - I	4
5	44	BCAOL506	LIBERAL STUDIES - II	1
				17

6	45	BCAOL601	PROJECT STAGE II	6
6	46	BCAOL602X	ELECTIVE - IV	4
6	47	BCAOL603X	ELECTIVE - V	4
				14
			Total Programme Credits	**128**

ELECTIVES			
ELECTIVE - I			
SEM	**SN.**	**Course Code**	**TITLE OF THE COURSE**
4	1	BCAOL4051	LINUX OPEN SOURCE SYSTEM
4	2	BCAOL4052	DEVOPS – I
4	3	BCAOL4053	IOT
4	4	BCAOL4054	PROBABILITY AND STATISTICS USING R
4	5	BCAOL4055	COMPUTER GRAPHICS AND MULTIMEDIA

ELECTIVE – II			
SEM	**SN.**	**Course Code**	**TITLE OF THE COURSE**
5	6	BCAOL5021	DATA MINING
5	7	BCAOL5022	MOBILE PROGRAMMING
5	8	BCAOL5023	SOFTWARE PROJECT MANAGEMENT
5	9	BCAOL5024	BLOCKCHAIN TECHNOLOGY

ELECTIVE – III			
SEM	**SN.**	**Course Code**	**TITLE OF THE COURSE**
5	10	BCAOL5031	DATABASE ADMINISTRATION
5	11	BCAOL5032	BIG DATA ANALYTICS
5	12	BCAOL5033	OPENSOURCE DEVELOPMENT USING PHP AND MYSQL
5	13	BCAOL5034	FULL STACK DEVELOPMENT

ELECTIVE – IV			
SEM	**SN.**	**Course Code**	**TITLE OF THE COURSE**
6	14	BCAOL6021	MACHINE LEARNING
6	15	BCAOL6022	GAME PROGRAMMING
6	16	BCAOL6023	ETHICAL HACKING
6	17	BCAOL6024	ADVANCE DATABASE(NOSQL)

ELECTIVE – V			
SEM	**SN.**	**Course Code**	**TITLE OF THE COURSE**
6	18	BCAOL6031	ARTIFICIAL INTELLIGENCE
6	19	BCAOL6032	ANIMATIONS
6	20	BCAOL6033	SOFT COMPUTING
6	21	BCAOL6034	CLOUD COMPUTING

5.3 Programme detailed syllabus

Semester: I

Course Name: MATHEMATICS - I

Credits: 4

Course Description:

This course provides the mathematical foundations essential for computing. Topics include set theory, logic, matrices, linear algebra, functions, and combinatorics. Learners will study discrete mathematics concepts such as permutations, combinations, and graph theory, which are pivotal for computer algorithms and structures. The course develops analytical thinking and problem-solving skills by applying mathematical techniques to real-world scenarios. By the end, learners will be equipped to handle advanced topics in algorithms, data analysis, and software development.

Course Objectives:

- Comprehend the foundational concepts of mathematical thinking, logic, and set theory to analyse problems systematically.
- Utilise functions, relations, and graph theory to model and solve real-world scenarios effectively
- Assess various graph properties, relationships, and statistical data using advanced mathematical and analytical techniques.
- Develop logical proofs, function compositions, and mathematical models to address complex computational and theoretical challenges.
- Integrate concepts from probability, calculus, and combinatorics to design solutions for multi-faceted problems across diverse disciplines.

Course Outcomes:

At the end of the course, the learner will be able to:

- Make use of mathematical logic and reasoning to solve problems
- Apply set theory and functions to model and solve problems

- Interpret the various graph theory concepts
- Apply counting techniques and use basic probability concepts
- Use calculus to calculate the rate of change and area under a curve.

Unit 1	Mathematical Thinking: Introduction to logic in computing, mathematical thinking, critical analysis, Propositional logic, Predicate logic, Inference and proof techniques, basic number theory and geometry
Unit 2	Set Theory: representations, operations, Venn diagram, cartesian product, power sets
Unit 3	Set Theory 2: Relations — Types of relations - reflexive, symmetric, transitive, antisymmetric, Equivalence relation, partial and total orderings
Unit 4	Functions: Types of functions – injective, surjective, bijective, function composition and inverse functions, Function plotting, Special functions – polynomials, exponential functions, logarithmic functions, trigonometric functions
Unit 5	Graph theory 1: graphs, vertices, edges, Types of graphs - undirected, directed, weighted, adjacency matrix, adjacency list, incidence matrix, Basic properties of graphs – degree, paths, walks, cycles, connectedness and components, Subgraphs, induced subgraphs and spanning subgraphs
Unit 6	Graph theory 2: Special types of graphs – complete graphs, bipartite graphs, trees and forests, planar graphs, Graph coloring – vertex coloring, chromatic number, Eurerian paths and circuits, Euler's theorem, Hamiltonian paths and circuits, Graph isomorphism (definition only)
Unit 7	Probability and statistics 1: Combinatorics – Basic counting principles, permutations and combinations, Binomial theorem, Pigeonhole principle, inclusion-exclusion principle
Unit 8	Statistics: Classification of Statistics, Descriptive Statistics, Mean and its types, Median, Mode, Measures of Dispersions, Variance and Standard Deviation
Unit 9	Probability and statistics 2: Sample space, events, probability axioms, conditional probability and independence, Bayes theorem — more stress on examples
Unit 10	Calculus 1: Definition of limits and continuity – Differential calculus – basic formulas, Techniques – power rule, product rule, quotient rule, chain rule, Tangent and rate of change
Unit 11	Calculus 2: Integral calculus – Basic formulas, Techniques – substitution, integration by parts, definite integrals, area under curves.
Unit 12	Matrices and Determinants: Definitions and Examples, Types of matrices, Operations on matrices, Invertible matrices, Determinants, Properties of determinants.

TEXTBOOKS:

- Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.

- Dr. P .R .Vittal : Mathematical Foundation, 2002, Margham Publications, ISBN-10 : 9383242299, ISBN-13 : 978-9383242290.

REFERENCE BOOKS:

- Shanti Narayan, "Differential Calculus", S Chand & Company.
- Shanthi Narayan, "Matrices and determinant", S. Chand and Company LTD
- B.S. Grewal, "Elementary Engineering Mathematics", 34th Ed. Delhi Khanna Publishers.
- Das BC and Mukherjee, Differential Calculus, Calcutta, U.N. Dhar Publishers.

Course Name: FUNDAMENTALS OF PROGRAMMING

Credits: 3

Course Description:

This course introduces the foundational concepts of programming using C. Learners will learn about data types, control structures, functions, arrays, pointers, and file handling. Emphasis is placed on problem-solving, algorithm development, and debugging techniques. The course includes hands-on practice implementing modular programs, structured coding practices, and effective memory management using pointers. By the end of the course, learners will have a solid foundation in programming logic, structured development, and program efficiency, equipping them to develop small-scale applications in C.

Course Objectives:

- Discuss the basic approaches for analysing the problem
- Learn the C language concepts required for building C program
- Develop essential skills like pointers to construct programs.
- Implement dynamic memory allocation and C language concepts required for building real-world applications.

Course Outcomes:

At the end of the course, the learner will be able to:

- Analyse the programming concept using algorithms and flowcharts.
- Apply basics of C programming and develop logic to write a program
- Develop the idea of arrays and strings and apply them in a C program
- Outline the concepts of functions and implement as a recursive function and dynamic memory allocation in structures and unions

Unit 1	Introduction to Problem Solving using Algorithms and Flowchart: Key features of Algorithms: Sequence, Decision, Repetition with examples, Understanding algorithms, Algorithm examples, Flowchart symbols and creation
Unit 2	Introduction to C Programming: History of C, Structure of C program, Keywords, Tokens, Character set, Identifiers, Data Types, Variables, Constants, Input/Output statements, Operators, Expressions, Type conversions, Conditional Branching Statements, Operators: Arithmetic, relational, logical, bitwise, conditional operators, Expressions: Precedence, Evaluation, Type conversions, Conditional statements: if, if-else, else-if ladder, switch
Unit 3	Iterative Statements (Loops): while, for, do-while statements, Loop examples, Nested loops, break, continue, goto statement, Loop constructs and syntax, Examples of loops in programs, Controlling loops with break, continue, goto
Unit 4	Arrays and Strings: Introduction, declaration & initialisation, reading and writing elements, Operations on arrays; Strings: Definition, declaration, initialisation, representation, string handling functions: Arrays: Traversal, searching, sorting, Two-dimensional arrays, Strings: Character arrays, string functions (strlen, strcpy, strcmp, strcat, etc.), character handling functions (toupper, tolower, isalpha, etc.)
Unit 5	Functions: Definition and declaration, Built-in functions and User-defined functions, Categories of functions, array as function argument, Scope and Lifetime of variables, Recursion: Function prototype, Passing arguments to functions, Return values, Recursive functions, Variable scope (local, global), Static variables
Unit 6	Pointers: Definition, declaration, and initialisation of pointers, Accessing values using pointers, pointers and arrays, Call-by-value and call-by-reference: Pointer arithmetic, Pointers to arrays and arrays of pointers, Passing pointers to functions, Dereferencing pointers, Difference between call-by-value and call-by-reference
Unit 7	Structures: Purpose and usage of structures, Declaration of structures, Assignment with structures, Arrays of structures, Nested structures: Defining structure types, Accessing structure members, Using structures with functions, Arrays of structures, Nested structures (structures within structures)
Unit 8	Unions and Applications: Declaration and initialisation, Difference between structures and unions; Applications: Learner and employee database implementation using structures: Defining unions, Accessing union members, Memory allocation in unions vs structures, Implementing practical applications (e.g., databases) using structures and unions
Unit 9	Memory Allocation and Preprocessor: Dynamic memory allocation, memory allocation functions (malloc, calloc, realloc, free), Macro substitution: Understanding heap memory, Using malloc, calloc for dynamic arrays, Resizing memory with realloc, Releasing memory with free, Preprocessor directives (#define, #include, macros)
Unit 10	Files: Defining, opening, and closing files, Character I/O: File pointers (FILE *), Modes of opening files (r, w, a, etc.) - Reading and writing characters and strings to files (fgetc, fputc, fgets, fputs) - Closing files (fclose)

TEXTBOOKS:

- Brian W. Kernigham and Dennis M. Ritchie, "The C Programming Language", 2nd Edition, PHI, 2012.
- R S Bichkar, Programming with C ,Universities press ,2012.
- Yashavant Kanetkar, Let Us C: Authentic guide to C programming language - 19th Edition, BPB Publications, 2022.

REFERENCE BOOKS:

- E Balaguruswamy , Programming in ANSI C, McGraw Hill Education(India) private limited, 9th Edition, 2024.
- Behrouz A. Forouzan, Richard F. Gilberg, "Computer Science - A Structured Approach Using C", Cengage Learning, 2007.
- Vikas Gupta, "Computer Concepts and C Programming", Dream tech Press 2013.Marchand& B: Vardharajan: An introduction to Marketing, Vikas Publishing House, New Delhi.
- Maurice & Modell & Larry Rosenberg: Marketing: Prentice Hall of India Ltd. New Delhi.
- Mohammad Amanatullah: Principles of Modern Marketing, Kalyani Publications New Delhi.
- Dr. C. N. Sontakki: Marketing Management, Kalyani Publications, New Delhi.

Course Name: COMPUTER ORGANIZATION

Credits: 3

Course Description:

This course covers the fundamentals of computer architecture, components, and operations. Learners will explore digital logic circuits, CPU architecture, memory systems, I/O mechanisms, and assembly language programming. Emphasis is given to understanding the internal workings of processors, instruction cycles, and data representation techniques. By the end of the course, learners will understand how computers process instructions, organise data, and achieve efficiency. The practical exposure prepares learners to design basic computing systems and analyse hardware-software interactions.

Course Objectives:

- Comprehend an overview of basic digital design techniques, the basic structure of a digital computer and its organisation.
- Identify the different number systems and codes.
- Discuss the digital representation of data in a computer system.
- Describe the general concepts in digital logic design, including logic elements and their use in combinational and sequential logic circuit design.

Course Outcomes:

At the end of the course, the learner will be able to:

- Convert various codes and number systems used in digital communication and computer systems
- Compare logic families and design efficient converting circuits for logic gates
- Simplify digital circuits using mapping, logical tools, and mathematical methods
- Apply analog and digital electronics knowledge for signal conversion and real-world memory device applications

Unit 1	Digital Computers and Software: Concept of Digital Computer, Types of Software: Definition and characteristics of digital computers, Components of a digital computer system, System software, Application software, Utility Software, Compilers, Interpreters, Assemblers, Linker, Loader
Unit 2	Bus Structure: System Bus Structure, Data Bus, Address Bus, Control Bus, Bus architecture (Single bus, Multiple bus structures), Bus communication protocols, Bus arbitration and timing
Unit 3	Number Systems: Number systems & conversions Signed Binary Representation, Decimal, Octal, Hexadecimal, Binary number systems, Converting between number systems, Binary arithmetic, Representation of negative numbers (Sign-magnitude, 1's complement, 2's complement)
Unit 4	Logic Gates and Boolean Simplifications: Logic Gates, Boolean Algebra and Simplifications, Basic logic gates (AND, OR, NOT), Universal gates (NAND, NOR), Other gates (XOR, XNOR), Boolean algebra principles, De Morgan's Theorems, Simplification techniques
Unit 5	Minimisation Techniques: SOP/POS, Karnaugh Map, Standard forms of Boolean expressions, Minimization using Karnaugh Maps for 2, 3, 4 variables, Quine-McCluskey method (overview)
Unit 6	Circuit Design: Codes, Arithmetic Operations, Combinational Circuits Design, Designing combinational circuits (Adders, Subtractors), Codes (BCD, Gray code, Excess-3), Arithmetic operations in digital circuits, Implementation of logic functions using gates
Unit 7	Combinational Circuits: Multiplexers, De-multiplexers, Decoders, Multiplexer (MUX) design and applications, De-multiplexer (DEMUX), Decoders and encoders, Implementing logic functions using MUX and DEMUX
Unit 8	Sequential Circuits: Sequential Circuit Design, Counters, Registers, Flip-flops (SR, JK, D, T), Designing counters (Asynchronous, Synchronous), Shift registers, Timing diagrams and state tables
Unit 9	Input and Output Devices and Memory Systems: Types of I/O devices (Input devices, Output devices), Memory hierarchy, RAM, ROM, Cache memory, Memory addressing
Unit 10	Interrupts, Instruction Cycle: Addressing Modes, Interrupt types and handling, CPU instruction cycle (Fetch, Decode, Execute), Addressing modes (Immediate, Direct, Indirect, Indexed)

TEXTBOOKS:

- M. Morris Mano, “Digital Logic and Computer Design”, Pearson Education, 2017.
- Malvino. D. Leach, “Digital Principles and Applications”, 7th Edition, Tata McGraw Hill.
- Floyd, “Digital Principles”, Pearson Education, 2002.
- Computer Fundamentals, by Priti Sinha, Pradeep K., Sinha, BPB Publications 8th

edition 2020.

REFERENCE BOOKS:

- Carl Hamacher, Z Varnesic and S. Zaky: Computer Organization, 6th Edition, McGraw Hill, 2023.
- W. Stalling, "Computer Organization and Architecture", 11th Edition, Pearson Education India, 2022.

Course Name: WEB PROGRAMMING

Credits: 4

Course Description:

This course provides a comprehensive understanding of the Internet and web development fundamentals. It explores the history of the Internet, web evolution phases (Web 1.0, Web 2.0, Web 3.0), and key protocols like TCP/IP, HTTP, and HTTPS. Learners will learn about web servers, browsers, URL structures, MIME types, and search engines. Core web technologies are introduced, including HTML for document structure, CSS for styling, and JavaScript for interactivity. Advanced topics like responsive design, DOM manipulation, error handling, and form validation are also covered. Learners will develop essential skills to build and manage dynamic, responsive web applications by the end.

Course Objectives:

- Discuss the foundations of web page design using HTML & CSS
- Design Client-Side programs using JavaScript
- Describe the Web page sessions and XML format
- Explain the fundamentals of JavaScript
- Build web pages using JavaScript

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss WWW, web servers
- Describe the foundations of web page design using HTML
- Design a web page and identify its elements and attributes using CSS
- Identify the fundamentals of JavaScript
- Build web pages using JavaScript

Unit 1	Introduction to Internet and its History: Internet, History of Internet, Evolution of web: Definition of the Internet, Milestones in Internet development, Phases of web evolution (Web 1.0, Web 2.0, Web 3.0)
Unit 2	Internet Protocols, World Wide Web (WWW):- Overview of TCP/IP, UDP, Functioning of the World Wide Web
Unit 3	Web Servers, Web Browsers: Types of Web Servers (Apache, IIS, Nginx), Popular Web Browsers and their features\
Unit 4	URL, MIME Types, and Search Engines: Structure of a URL, Common MIME types, Working of Search Engines
Unit 5	HTTP, HTTPS, and Advanced Web Technologies: HTTP, HTTPS, Security, Web Sockets, HTTP Streams, Basics of RPC: HTTP Protocol Methods, Secure HTTP (HTTPS) and SSL/TLS, Introduction to Web Sockets and real-time communication, Understanding HTTP Streams, Basics of RPC
Unit 6	Basic Syntax and HTML Document Structure: Doctype declaration, HTML tags and nesting, Head and Body sections
Unit 7	HTML Tags and Elements: Text formatting tags, Lists, Links, and Images, Tables and Forms
Unit 8	Frames and HTML DOM: Using Frames and Iframes, Introduction to the HTML Document Object Model
Unit 9	Introduction to CSS and the Box Model: Inline, Internal, and External CSS, Components of the Box Model (Margin, Border, Padding, Content)
Unit 10	Advanced CSS Topics: Selectors, Positioning, Responsive Design, CSS Selectors and specificity, Positioning elements (static, relative, absolute, fixed, sticky), Media queries and Responsive Web Design
Unit 11	JavaScript Basics: Introduction to JavaScript, Types of Variables, Expressions, Data types and variables, Operators and expressions (boolean, arithmetic)
Unit 12	Control Structures and Functions: Conditions (if-else, switch), Loops (while, for), Functions (parameters, arrow function), Conditional statements, Looping constructs, Function declaration and usage, Arrow functions
Unit 13	Error handling and Regular Expressions: try-catch-finally, Regular Expressions: Try-Catch-Finally blocks, Creating and using Regular Expressions
Unit 14	Form Handling in JavaScript and HTML Documents: Accessing form elements, Validating form data, Submitting forms via JavaScript

TEXTBOOKS:

- Robert W. Sebesta: Programming the World Wide Web-8th Edition, Pearson Education, 2015.
- David Choi, “Full-Stack React, TypeScript, and Node”, packt, 2020.

REFERENCE BOOKS:

- Robin Nixon, "Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5",4thEdition, O 'Reilly Publications, 2018 (ISBN:978-9352130153).
- Luke Welling, Laura Thomson, "PHP and MySQL Web Development", 5th Edition, Pearson Education, 2016 (ISBN:978-9332582736).
- Nicholas C Zakas, "Professional JavaScript for Web Developers", 4th Edition, Wrox/Wiley India, 2015 (ISBN: 0764579088).
- David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", 1st Edition, O 'Reilly Publications, Shroff Publishers & Distributors Pvt Ltd, 2014.

Course Name: ACCOUNTING AND FINANCIAL MANAGEMENT

Credits: 4

Course Description:

This course provides a foundational understanding of financial accounting, its principles, and practical applications. Learners will explore accounting mechanics, the double-entry system, and formats for preparing journals, ledgers, subsidiary books, and trial balances. Key concepts include depreciation (SLM & WDV methods), financial statements with adjustments, and bank reconciliation statements. Topics like consignment accounts, bills of exchange, and the sale of goods on an approval or return basis are introduced. The course also uses AI-based fraud detection techniques to examine modern aspects like financial reporting, inflation accounting, and forensic accounting. By the end, learners will gain essential accounting and reporting skills.

Course Objectives:

- Discuss the accounting principles and GAAP
- Prepare journals, ledgers, and statements
- Analyse depreciation and adjustment entries
- Evaluate financial performance and reports
- Create records for specialised transactions

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain the fundamentals of accounting
- Apply accounting mechanics and processes
- Analyse financial adjustments and statements
- Solve bank reconciliation and consignment problems
- Demonstrate knowledge of advanced accounting

Unit 1	Meaning & Scope of Accounting: Introduction, Meaning & Scope of Financial Accounting, Meaning and Nature of Business Transaction, Introduction to Basic Elements of Financial Accounting, Accounting Principles: Accounting Concepts, Principles and Conventions and Assumptions, Generally Accepted Accounting Principle (GAAP)
Unit 2	Accounting Mechanics: Principles of Double Entry, Accounting Equation
Unit 3	Accounting Process: Formats and Preparation of Journal Entries, Formats and Preparation of Ledger Accounts, Formats and Preparation of Subsidiary Books, Formats and Preparation of Triple Column Cash Book and Trial Balance
Unit 4	Depreciation: Methods for Calculating Depreciation (SLM & WDV)
Unit 5	Financial Statements: Formats and preparation of Income Statement & Balance Sheet of a Sole Trader with Adjustments
Unit 6	Adjustment Entries: Outstanding and Prepaid expenses, Bad Debts, Depreciation, Treatment of closing stock, Provision of Doubtful debts and taxation, Proposed Dividend and reserves tax provision, dividend, and reserves, finding EPS
Unit 7	Introduction to Bank Reconciliation Statement: Meaning, Scope and Objectives of Bank Reconciliation Statement
Unit 8	Pass book and Cash book: Meaning and Format of Pass book and Cash Book
Unit 9	Bank Balances and Problem Questions: Concept & Formats of Bank Balances: Ordinary & Overdraft, Simple Problems on Cash Book and Pass Book
Unit 10	Consignment Accounts: Introduction to Consignment accounts: Concept & Journal entries in the books of consignor and consignee, Types of commisions in consignment
Unit 11	Bills of Exchange and Promissory Notes: Introduction and Parties to Bills of Exchange, Journal Entries and Promissory Notes
Unit 12	Sale of goods on approval or return basis: Introduction, Features, Books maintained by business entity, Problems on Sale of goods on approval or return basis
Unit 13	Financial Reporting and Inflation Accounting: Meaning, Types, Importance and Purpose of Financial Reporting, Meaning, Advantages and Disadvantages of Inflation Accounting
Unit 14	Forensic Accounting and Fraud Detection: Meaning and Types of Forensic Accounting, Meaning, Types and AI-based techniques, Application and Best Practices of Fraud Detection.

TEXTBOOKS:

- Narayanswami, “Financial Accounting: A Managerial Perspective”, PHI, 7th Edition, 2022.

REFERENCE BOOKS:

- Mukherjee, “Financial Accounting for Management”, TMH, 1st Edition.
- Gupta Ambrish, “Financial Accounting for Management”, Pearson Education, 6th Edition, 2018.

- Chowdhary, Anil, “Fundamentals of Accounting and Financial Analysis”, Pearson Education, 2007, ISBN: 8131776077, 9788131776070.

Course Name: ENGLISH IN PRACTICE

Credits: 3

Course Description:

This course enhances learners' grammar, vocabulary, and communication skills essential for academic and professional success. Topics include grammar essentials like tenses, subject-verb agreement, and word formation, as well as homonyms, homophones, synonyms, and antonyms to expand vocabulary. It develops listening and speaking skills through active listening, pronunciation, and public speaking practice. Learners learn group discussion techniques, data interpretation using charts and reports, and academic writing skills, including referencing (APA, MLA, Chicago styles) and plagiarism avoidance. Additionally, the course covers reading comprehension strategies, creative writing (essays, stories, poetry), and personal narratives, ensuring a well-rounded mastery of language proficiency.

Course Objectives:

- Enable learners to improve their lexical and grammatical competence
- Enhance their communicative skills
- Equip learners with oral and appropriate written communication skills
- Inculcate learners with employability and job search skills

Course Outcomes:

At the end of the course, the learner will be able to:

- Enhance speaking and vocabulary skills to communicate confidently and fluently in daily discussions
- Develop listening and reading comprehension abilities to understand instructions and improve performance
- Apply precise communication techniques for practical expression across diverse topics and scenarios
- Improve writing and problem-solving skills to produce quality work and convey ideas clearly through technical and creative writing.

Unit 1	Grammar Essentials: Introduction to Grammar Essentials, Types of Tenses in English: Present, Past, Future (Simple, Continuous, Perfect, Perfect Continuous), Concord (subject-verb agreement): Definition & Types, Word Formation: Definition & Concept, Types of Word Formation (Derivation, Compounding, Blending, Acronym, Borrowing, Coinage, Back-formation, Re-duplication, Clipping), Kinds of Word formation (prefixes, suffixes, roots, compounds), Sentences: Structure, Types & Application
Unit 2	Homonyms and Homophones: Introduction to Homonyms and Homophones: Concept & Understanding, Homonyms: Definition, Functions & Examples, Homophones: Definition, Functions & Examples, Common Word Choice Confusions: Avoiding Errors in Context, Spelling, and Writing, Synonyms and Antonyms: Expanding vocabulary through meaning-based word study
Unit 3	Listening Skills: Introduction to Listening Skills, Active Listening Techniques: Focused Listening, Paraphrasing, and Summarising, Identifying Sounds: Differentiating Between Vowel and Consonant Sounds, Note-Taking During Listening: Methods for Capturing Key Points Effectively, Listening for Specific Information: Focus on Key Details, Instructions, and Tone in Conversation
Unit 4	Speaking Skills: Introduction to Speaking Skills, Common Pronunciation Errors: Individual Sounds, Intonation Patterns, Accent Reduction: Tips for Neutralising Regional Accents, Speaking Practice: Role-plays, Everyday Conversations, Peer Feedback, Public Speaking Techniques: Using Body Language, Voice Modulation, and Audience Engagement
Unit 5	Process Description in Speaking: Introduction to Process Description in Speaking, Describing the Working of Machines: Vocabulary for Technical Descriptions, Explaining Manufacturing Processes: Step-by-step Explanations, Sequence Markers, Process Presentation Techniques: Clarity in Delivery, Use of Visual Aids, Describing Everyday Processes: Cooking Recipes, Task Instructions, and Simple Processes in Daily Life
Unit 6	Group Discussion Techniques: Introduction to Group Discussion Techniques, Role of Moderators: Facilitating Discussion and Managing Group Dynamics, Structured Discussions: Introducing Topics and Summarising Viewpoints, Argument Formation: Supporting Opinions with Facts and Handling Counter-arguments, Team Dynamics and Cooperation: Understanding Roles in Teamwork and Encouraging Collaboration.
Unit 7	Data Interpretation in Writing: Introduction to Data Interpretation in Writing, Interpreting Flow Charts and Diagrams: Understanding Processes and Relationships, Bar Charts and Pie Charts: Analysing Data Distribution and Trends, Drawing Conclusions from Data: Summarising and Presenting Findings, Writing Analytical Reports: Presenting Data-supported Conclusions with Clarity
Unit 8	Academic Report Writing & Referencing Skills: Introduction to Academic Report Writing, Referencing Skills: APA, MLA, and Chicago Styles of Citation, Avoiding Plagiarism: Paraphrasing, Quoting, and Acknowledging Sources, Report Structure: Introduction, Body, Conclusion, and Appendices, Abstract Writing: Condensing Report Content into Concise Abstracts
Unit 9	Reading Comprehension & Answering Questions: Introduction to Reading Comprehension, Skimming and Scanning Techniques: Rapid Reading Strategies to Find Key Information, Answering, Questions from Texts: Identifying Main Ideas and Supporting Details, Analysing Different Text Types: Narrative, Descriptive &

	Analytical Texts, Inference and Interpretation: Reading between the Lines to Understand Implied Meaning
Unit 10	Appreciation of Creative Writing: Introduction to Creative Writing, Essay Writing: Structuring Essays, Forming Arguments & Writing Conclusions, Story Creation: Crafting Plots, Character Development & Narrative Techniques, Poetry Appreciation & Writing: Understanding Poetic Devices and Writing Original Poetry, Writing Personal Narratives: Crafting Reflective & Engaging Personal Stories

TEXTBOOKS:

- Dhanavel, S.P. "English and Communication Skills for Learners of Science and Engineering". Orient Blackswan Pvt. Ltd., 2009 Print.
- Meenakshi Raman and Sangeetha Sharma. "Technical Communication- Principles and Practice". 3rd Edition, Oxford University Press, 2015.
- Murphy R. "English Grammar in Use", Cambridge University Press, (South Asian Edition) 2023.

REFERENCE BOOKS:

- Day. R A. "Scientific English: A Guide for Scientists and Other Professional", 2nd Edition, Hyderabad: Universities Press, 2000 Print.
- Ashraf Rizvi M. "Effective Technical Communication", McGraw Hill Education, 2017 Print.
- Eastwood J. "Oxford Practice Grammar", Oxford University Press, 11th Impression, 2003 Print.
- Swan M and Walter C. "Oxford English Grammar Course Advanced Learner's Book", Oxford University Press, Revised Edition, 2019 Print.
- Dale, Carnegie. "The Quick and Easy Way to Effective Speaking", JAICO Publishing House, 2019 Print.

Course Name: CONSTITUTION OF INDIA AND PROFESSIONAL ETHICS

Credits: 1

Course Description:

This course provides an in-depth understanding of the structure, principles, and significance of the Indian Constitution. It explores the Constitution's making, sources, and salient features, including the Preamble, Fundamental Rights, Duties, and Directive Principles. Learners will study the Union and State Governments, focusing on the roles and powers of the President, Prime Minister, Parliament, Governors, and State Legislatures. It also examines the Indian Judiciary, covering the Supreme Court and High Courts' structure and jurisdiction. Topics include federalism, local governance (73rd & 74th Amendments), the Election Commission, and citizen-oriented measures like RTI and PIL, highlighting their importance.

Course Objectives:

- Outline basic information about the Indian constitution
- Identify individual roles and ethical responsibility towards society

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss state and central policies, fundamental duties, electoral processes, and special provisions
- Analyse local governance and professional ethics

Unit 1	Constitution – Structure and Principles: Introduction to Constitution, Meaning and Importance of Constitution, Indian Constitution –Making & Sources, Preamble and Salient features of Indian Constitution, Fundamental rights; fundamental duties; directive principles
Unit 2	Government of the Union and States: Introduction to Government of the Union and States, Government of the Union: President of India – election and powers; prime minister and council of ministers; Lok Sabha – composition and powers; Rajya Sabha – composition and powers, Government of the States: Governor – powers; Chief Minister and Council of Ministers; Legislative Assembly – composition and powers; Legislative Council – composition and powers.
Unit 3	The Judiciary: Introduction to Judiciary, Features of judicial system in India, Supreme court –Structure and Jurisdiction, High court – Structure and Jurisdiction
Unit 4	Administrative Organisation and Constitution: Introduction to Constitution, Federalism in India, Features

	Local Government -Panchayats –powers and functions,73rd and 74th amendments, Election commission – organisation and functions, citizen-oriented measures – RTI and PIL – provisions and significance

TEXTBOOKS:

- Brij Kishore Sharma,"Introduction to the Constitution of India", PHI Learning Pvt. Ltd., 11th Edition, New Delhi, 2022.
- Durga Das Basu: "Introduction to the Constitution on India", (Learners Edn.) PrenticeHall, 19th / 20th Edn., 2001.

REFERENCE BOOKS:

- M.V.Pylee, "An Introduction to Constitution of India", Vikas Publishing, 2002.

Semester: II

Course Name: MATHEMATICS - II

Credits: 4

Course Description:

This course introduces the fundamentals of probability and statistics, starting with sample spaces, events, and foundational rules like addition, multiplication, and Bayes' theorem. It explores random variables, counting techniques, and discrete and continuous probability distributions, including binomial, Poisson, standard, and exponential distributions. Learners will learn statistical methods for data classification, graphical representation, correlation (Pearson and Spearman), and regression analysis. Key topics include statistical inference, hypothesis testing, sampling theory, and the Central Limit Theorem. The course also covers statistical experiment design, focusing on ANOVA, linear hypotheses, and time-series analysis with practical applications for real-world scenarios.

Course Objectives:

- Discuss the probability concepts and distributions.
- Apply statistical methods for data analysis
- Evaluate relationships using correlation, regression
- Test hypotheses and estimate parameters
- Design experiments and analyse variance

Course Outcomes:

At the end of the course, the learner will be able to:

- Apply probability concepts and rules
- Analyse probability distributions
- Use statistical methods for data analysis
- Perform regression analysis and hypothesis testing
- Design and analyse experiments

Unit 1	Probability 1: Introduction, Sample Space, Events, Probability, The Addition Rule, The Multiplication Rule, Illustrations, Bayes' Formula
Unit 2	Probability 2: Counting Techniques, Random variables, Discrete random variables, Density functions, Joint density functions, Marginal and conditional distributions, Continuous random variables
Unit 3	Probability Distribution 1: Discrete variables, Expectation, Moments, Moment generating function, Binomial distribution, Poisson distribution
Unit 4	Probability Distribution 2: Continuous variables, Moments, Moment generating function, Rectangular distribution, Normal distribution, moments, normal approximation to binomial, Exponential distribution, Chi-Square distribution
Unit 5	Statistical Methods: Introduction, Data, Classification of data, graphical representation of empirical distributions, empirical moments
Unit 6	Correlation: definition, Pearson correlation, Spearman rank correlation
Unit 7	Regression: linear, least squares method, regression coefficient
Unit 8	Statistical Inference 1: Statistical inference, estimation, testing hypothesis, two types of error
Unit 9	Statistical Inference 2: Sampling theory, random sampling, properties of E – Optimal weighing, Expectation formulas for sums and means
Unit 10	Statistical Inference 3: Distribution of mean from a normal distribution, applications Central limit theorem Testing a mean, testing the difference of two means
Unit 11	Statistical Design in Experiments 1: Randomization, replication and sensitivity, analysis of variance, linear hypothesis model, application
Unit 12	Statistical Design in Experiments 2: Two-way anova, application Time-series, components and moving averages (basics),Practice problems

TEXTBOOKS:

- Hoel, PG: Introduction to Mathematical Statistics (1962), John Wiley and Sons, New York.
- Milton. J. S. and Arnold. J.C., "Introduction to Probability and Statistics", Tata McGraw Hill, 4th Edition, 2007.
- Johnson. R.A. and Gupta. C.B., "Miller and Freund"s Probability and Statistics forEngineers", Pearson Education, Asia, 7th Edition, 2007.

REFERENCE BOOKS:

- Chung K.L. (1983): Elementary Probability Theory with Stochastic Process, Springer / Narosa.
- Kossack, C.F. and Hensschkec, C.I., Introduction to Statistics and Computer

Programming, Tata McGraw-Hill, New Delhi.

- Wayne L. Winston: Operations Research. Thomson, India 4th edition, 2004.

Course Name: OBJECT ORIENTED PROGRAMMING USING JAVA

Credits: 3

Course Description:

This course introduces the principles of Object-Oriented Programming (OOP) and its advantages over procedural programming paradigms. Learners will explore the fundamentals of Java, including its structure, data types, variables, operators, and input/output mechanisms. Core OOP concepts such as classes, objects, inheritance, polymorphism, abstraction, and encapsulation are covered in detail. Key topics include arrays, strings, exception handling, and multithreading, with practical applications to manage errors, create packages, and handle multiple threads. The course also discusses Java Virtual Machine (JVM), package creation, and synchronisation, providing learners hands-on experience to build efficient, modular, and secure Java applications.

Course Objectives:

- Discuss the basic data types, operators and control structures in Java
- Describe the basic handling of packages and interfaces in Java
- Apply the Java thread model to program Java applications
- Develop GUI applications using Java swings

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss OOP concepts and Java fundamentals
- Implement variables, arrays, and strings
- Use packages and exception handling
- Develop multithreaded applications

Unit 1	Introduction to OOP: Need for OO paradigm, comparison of OOP and POP, Benefits of OOPS.
Unit 2	Introduction to Java: History of Structure of C program, keywords, Tokens, Characterset, Identifiers, Data Types, Variables, Constants, Input / Output statements. Operators A First Simple Program, Parts of Java program, Salient features of Java, The Java Virtual Machine (JVM), Naming Conventions in Java, Applications of java, Basic oops concepts, Reading Input using scanner Class, Displaying Output using System.out.println(), Command Line Arguments. Installing JDK.
Unit 3	Variables: Data types, variables- Scope & lifetime of variables.
Unit 4	Arrays: Types of Arrays, array name, length
Unit 5	Strings: Creating Strings, String Class Methods, String Comparison. Identifiers, Key words, Operators, expressions, control statements, type conversion and casting
Unit 6	OOPs - Part 1: Classes & objects, Inheritance, Types of Inheritance Polymorphism: Method overloading, overriding
Unit 7	OOPs - Part 2: Abstract Method and Abstract Class, Encapsulation, constructor and destructor.
Unit 8	Packages and Interfaces: Package, Different Types of Packages, Creating Package and Accessing a Package, Introduction to Interfaces, Implementing Interfaces
Unit 9	Exception Handling: Errors in Java P.rogram, Exceptions, throws Clause, throws Clause, Types of Exceptions, types of exception handling- Built in exceptions, custom exceptions, Default and user-defined exception handling mechanism.
Unit 10	Multithreading: Java Thread Model, what is thread, Thread life cycle, creating threads, multithreading, methods of multithreading. Synchronisation, Case study

TEXTBOOKS:

- Java: The complete reference, 13th Edition, Herbert Schildt, McGraw Hill, 2024, Osborne.
- Object Oriented Programming With Java By M. T. Somashekara, D. S. Guru, K. S. Manjunatha, PHI Learning, 2017.
- Programming with Java- A primer, 7th Edition, E.Balaguruswamy, TataMcGraw-Hill publishing company limited, New Delhi, 2023.

REFERENCE BOOKS:

- Java: How to Program, 11/e, Dietal, Dietal, PHI.
- Java 17 for Absolute Beginners, Iuliana Cosmina, 2021.
- T.Budd, “Understanding Object-Oriented Programming with Java”, Pearson Education, Updated Edition (New Java 2 Coverage), 2000.

DAYANANDA SAGAR UNIVERSITY

Course Name: DATA STRUCTURES

Credits: 3

Course Description:

This course explores data structures, including arrays, linked lists, stacks, queues, trees, and graphs. Learners will learn sorting, searching, and hashing algorithms focusing on complexity analysis and optimisation techniques. Practical assignments emphasise implementing algorithms to solve computational problems efficiently. By the end of the course, learners will be able to choose appropriate data structures, design algorithms, and analyse their performance, laying a foundation for advanced computing tasks.

Course Objectives:

- Discuss the concepts of data structures and complexities of algorithms
- Implement arrays, stacks, and queues
- Apply linked lists and tree algorithms
- Analyse advanced trees and search applications

Course Outcomes:

At the end of the course, the learner will be able to:

- Implement fundamental data structures
- Apply stack and queue concepts
- Develop linked list solutions
- Utilise tree algorithms and balancing techniques

Unit 1	Introduction and Algorithm Analysis: Primitive and Non-Primitive Data Structures, Algorithm Analysis and Complexity, Definition and examples of primitive data structures (int, float, char), Non-primitive data structures (arrays, lists, files), Time complexity, Space complexity, Asymptotic Notations (Big O, Omega, Theta)
Unit 2	Arrays: One-Dimensional, Two-Dimensional, Sparse Matrix Representation:, Declaration and initialisation of arrays Memory representation, Operations on arrays (insertion, deletion, traversal), Sparse matrices and their storage
Unit 3	Searching Techniques: List Searches, Sequential Search, Binary Search, Analysing Search Algorithms: Linear search algorithm, Binary search algorithm, Algorithm analysis and comparison, Applications of searching algorithms

Unit 4	Stacks and Queues: Stack Definition, Array Representation, Operations, Queue Definition, Representation, Operations, Stack applications: Expression evaluation, Infix to Postfix/Prefix conversion, Recursion, Tower of Hanoi, Queue applications: Job scheduling, Maze problem
Unit 5	Advanced Stack Applications: Detailed Case Studies: Expression evaluation algorithms, Conversion algorithms between infix, postfix, and prefix expressions, Implementation details of recursion using stack
Unit 6	Advanced Queue Applications: Detailed Case Studies: Implementation of circular queues, Job scheduling algorithms, Solving maze problems using queues
Unit 7	Linked Lists: Types and Representation of Linked Lists in Memory: Singly linked lists, Doubly linked lists, Circular linked lists, Memory allocation for linked lists, Operations: Traversing, searching, insertion, deletion
Unit 8	Applications of Linked Lists: Stack and Queue, Implementation using Linked Lists: Implementing stacks and queues using linked lists, Solving the Josephus problem using circular linked lists
Unit 9	Trees: Basic Terminology, Types of Trees, Operations: Definitions: Node, root, leaf, subtree, Binary trees, complete binary trees, binary search trees, threaded binary trees, Operations: Insertion, deletion, search, traversal (in-order, pre-order, post-order)
Unit 10	Tree Applications and Advanced Trees: Expression Evaluation, Game Trees, Efficient Binary Search Trees: AVL Trees, Red-Black Trees, Splay Trees, B-Trees: Evaluating arithmetic expressions using trees, Game tree concepts in AI, Balanced tree concepts, Implementing AVL trees, Understanding red-black trees and splay trees, Case study on B-Trees

TEXTBOOKS:

- A.M. Tannenbaum, Y Langsam, M J Augentien “Data Structures using C”, 1st Edition, Pearson, 2019.
- Ellis Horowitz, Susan Anderson-Freed, and Sartaj Sahni, “Fundamentals of Data structures in C”, 2nd Edition, Orient Longman, 2008.
- Brian. W. Kernighan, Dennis. M. Ritchie, “The C Programming Language”, 2nd Edition, Prentice-Hall, 1988.
- Dr.S.Nandagopalan, “Data structures and Applications”,1st Edition, Pearson, 2016.

REFERENCE BOOKS:

- Gilberg & Forouzan, “Data Structures: A Pseudo-code approach with C”, 2nd Edition, Cengage Learning,2014.
- Jean-Paul Tremblay & Paul G. Sorenson, “An Introduction to Data Structures with Applications”, 2nd Edition, McGraw Hill, 2013.

- R.L. Kruse, B.P. Learly, C.L. Tondo, “Data Structure and Program design in C”, 5th Edition, PHI ,2009.

Course Name: COMPUTER ARCHITECTURE

Credits: 4

Course Description:

This course thoroughly studies the 8086 microprocessor and its architecture, evolution, and functionality. Learners will explore pin configurations, timing diagrams, and minimum/maximum mode operations. The course addresses modes, data movement, arithmetic and logic instructions, and program control mechanisms such as loops, jumps, and subroutines. Advanced topics include modular programming, assembler directives, and interrupt-handling techniques. Additionally, it covers I/O interfaces, DMA operations, and peripheral devices like 8259 and 8255. Pipelining concepts, hazard mitigation, and modern processor case studies (e.g., Intel Core, ARM Cortex) are analysed to provide insights into current processor design and performance advancements.

Course Objectives:

- Discuss the 8086 architecture, pins, and modes
- Apply addressing modes and instructions
- Develop programs using assembly techniques
- Analyse interrupts, I/O interfaces, and controllers
- Evaluate pipelining concepts and modern processors

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the 8086 microprocessor architecture and operations
- Analyse instructions and programming techniques
- Apply modular and assembly programming
- Explore interrupts and I/O interfacing
- Examine pipelining and modern processors

Unit 1	Introduction to Microprocessor and Computer: Internal Architecture of 8086 Processor, Functional Block Diagram, Overview of Microprocessors, Evolution of 8086 Processor, Detailed Study of Functional Units, Register Organization
Unit 2	8086 Microprocessor Pins and Signals: Pins and Signals of 8086, Bus Cycles and Clock Signals, Pin Configuration and Functions, Signal Descriptions, Timing Diagrams, Clock Generation and Synchronization
Unit 3	Minimum and Maximum Modes of 8086: Working of Minimum Mode, Working of Maximum Mode, Mode Selection Pins, System Configuration in Minimum Mode, System Configuration in Maximum Mode, Comparison Between Modes
Unit 4	Addressing Modes in 8086: Addressing Modes, Immediate Addressing, Register Addressing, Direct Memory Addressing, Indirect Addressing Modes, Indexed and Based Addressing
Unit 5	Data Movement, Arithmetic, and Logic Instructions: Data Movement Instructions, Arithmetic Instructions, Logic Instructions, MOV, PUSH, POP, XCHG Instructions, ADD, SUB, INC, DEC, MUL, DIV Instructions, AND, OR, XOR, NOT Instructions, Flags Affected, Usage and Examples
Unit 6	Program Control and String Instructions: Program Control Instructions, String Instructions, JMP, CALL, RET Instructions, Conditional Jump Instructions (JE, JNE, JC, JNC, etc.), LOOP Instructions, String Instructions (MOVS, CMPS, SCAS, LODS, STOS), Direction Flag Manipulation - Repetition Prefixes (REP, REPE, REPNE), Usage and Examples
Unit 7	Modular Programming Techniques: Stacks, Subroutines, Macros, Stack Operations and Management, Defining and Calling Subroutines, Parameter Passing Mechanisms, Creating and Using Macros
Unit 8	Programming Techniques and Examples: Programming Techniques, Examples, Writing Efficient Code, Debugging and Testing Strategies, Sample Programs to Illustrate Concepts
Unit 9	Assembler Directives: Data Definition and Storage Allocation, Program Organization, Alignment, Program End, Value Returning Attribute, Procedure Definition, Macro Definition, Data Control, Branch Displacement, Header File Inclusion, Target Machine Code, Understanding Assembler Directives, Organizing Assembly Programs, Memory Allocation Strategies, Controlling Program Flow, Generating Machine Code
Unit 10	Interrupts and Interrupt Routines: Types of Interrupts (Hardware, Software, Exceptions), Interrupt Vector Table, Interrupt Handling Mechanism, Writing Interrupt Service Routines (ISRs)
Unit 11	I/O Interface and Programming: Fundamental I/O, Considerations, Data Transfer Schemes Programmed I/O, Interrupt I/O, DMA, System Bus Structure, Memory-Mapped I/O vs. I/O-Mapped I/O, Synchronous and Asynchronous Data Transfer, DMA Controller Operation, System Bus Standards (ISA, PCI, etc.), Programming Examples
Unit 12	Application of 8259 and 8255: Programmable Interrupt Controller (PIC) 8259, Programmable Peripheral Interface (PPI) 8255, Interfacing Techniques, Practical Applications

Unit 13	Pipelining Concepts and Implementation: Basic Concepts, Data Hazards, Instruction Hazards, Data Path and Control Considerations, Performance Considerations, Exception Handling, Introduction to Pipelining, Pipeline Stages, Types of Hazards (Structural, Data, Control), Techniques to Mitigate Hazards, Designing Data Paths for Pipelining, Control Unit Design, Evaluating Pipeline Performance, Handling Exceptions and Interrupts in Pipelines
Unit 14	Case Study - Latest Processor: Case Study of a Modern Processor Implementing Pipelining, Study of Intel Core Series or ARM Cortex Processors, Pipelining Implementation Details, Advanced Features (Out-of-Order Execution, Branch Prediction, etc.), Performance Analysis

TEXTBOOKS:

- Barry B Brey - The Intel Microprocessors, 2002, PHI.
- Douglas V. Hall- Microprocessors and digital systems, Gregg Division, McGraw-Hill, 1980.
- Microprocessor 8086: Architecture, Programming and Interfacing: Mathur Sunil, PHI Learning, 2010.
- Carl Hamacher, Computer Organization, 6th edition, McGraw-Hill, 2023.

REFERENCE BOOKS:

- YuChang Liu & Glenn A Gibson, “Microcomputer systems: the 8086\8088 Family: Architecture, Programming and design”, Prentice-Hall, 1984.
- Kenneth L. Short - Microprocessor and Programmed Logic, Pearson Education India, 2nd Edition, 2003.

Course Name: TECHNICAL COMMUNICATIONS

Credits: 3

Course Description:

This course focuses on essential writing and communication skills needed for academic and professional success. Learners learn to prepare effective abstracts and synopses, mastering their purpose, structure, and key components. Notice writing, formal letters, and technical paper drafting introduce clarity, conciseness, and appropriate tone in communication. The course covers minutes of meetings, memo writing, and email etiquette for professional settings. Learners also explore informal communication techniques, icebreakers, and overcoming barriers to communication. Practical skills such as CV and resume drafting, cover letter writing, and job applications are addressed to ensure readiness for professional opportunities. Real-world examples and exercises enhance learning.

Course Objectives:

- Develop effective abstracts, synopses, and notices.
- Master technical paper and formal letter writing
- Create professional resumes, CVs, and cover letters
- Write clear memos, emails, and meeting minutes

Course Outcomes:

At the end of the course, the learner will be able to:

- Develop technical writing skills for effective communication and accurate preparation of research data
- Enhance content quality by mastering technical writing techniques to strengthen professional connections
- Apply technical writing in real-life scenarios to foster responsibility, relationships, and a positive learning environment.

- Create professional documents such as resumes, CVs, and cover letters to showcase skills and unique qualities.

Unit 1	Preparation of Abstract and Synopsis: Understanding Abstracts, Purpose of Abstracts, Types of Abstracts, Writing Effective Abstracts, Understanding Synopses, Purpose and Structure of a Synopsis, Components of an Abstract, Tips for Writing Abstracts, Examples and Practice, Components of a Synopsis, Steps in Preparing a Synopsis
Unit 2	Notice Writing: Purpose of Notices, Format and Structure of Notices, Language and Tone in Notices, Examples of Notice Writing, Elements of a Notice, Do's and Don'ts in Notice Writing, Practice Exercises
Unit 3	Technical Paper Writing: Understanding Technical Papers, Structure of a Technical Paper, Writing Process, Audience Analysis, Introduction, Methods, Results, Discussion (IMRAD), Citation and Referencing Styles, Clarity and Conciseness, Common Pitfalls in Technical Writing
Unit 4	Minutes of the Meeting: Purpose of Meeting Minutes, Types of Meetings and Minutes, Structure and Format of Minutes, Best Practices in Minute Writing, Recording Attendance, Documenting Discussions and Decisions, Action Items and Follow-ups, Legal and Confidentiality Considerations
Unit 5	Formal Letters: Types of Formal Letters, Letters of Enquiry, Letters of Permission, Letters of Regret, Letters of Reconciliation, Letters of Complaint, Format and Structure, Language and Tone, Addressing and Salutations, Body of the Letter, Closing and Signature, Politeness and Professionalism
Unit 6	Informal Communication and Breaking the Ice: Importance of Informal Communication, Techniques for Breaking the Ice, Building Rapport in Groups, Encouraging Open Communication, Icebreaker Activities, Effective Use of Language in Informal Settings, Overcoming Communication Barriers, Fostering a Positive Learning Environment
Unit 7	Drafting Curriculum Vitae and Resumes: Differences between CV and Resume, Essential Components, Formatting and Presentation, Tailoring for Specific Jobs, Personal Information, Education and Qualifications, Work Experience, Skills and Achievements, References

Unit 8	Cover Letters and Job Applications: Purpose of Cover Letters, Structure and Content, Personalizing Applications, Writing Effective Job Applications, Addressing the Employer, Highlighting Relevant Experience, Expressing Interest and Fit, Follow-up Actions
Unit 9	Memo Writing: Purpose of Memos, Structure and Format, Language and Tone, Distribution and Confidentiality, Heading Components, Clear and Concise Messaging, Action-Oriented Language, Internal Communication Policies
Unit 10	Email Etiquette: Importance of Email Communication, Professional Email Structure, Netiquette Guidelines, Common Email Mistakes, Subject Lines, Greetings and Sign-offs, Replying and Forwarding, Managing Attachments, Privacy and Security Considerations

TEXTBOOKS:

- N. Krishnaswamy and T. Sri Raman. "Creative English for communication", Laxmi Publications Private Limited, 2022.
- Dhanavel, S.P. "English and Communication Skills for Learners of Science and Engineering". Orient Blackswan Pvt. Ltd., 2009 Print.
- Meenakshi Raman and Sangeetha Sharma. "Technical Communication- Principles and Practice". 3rd Edition, Oxford University Press, 2015.

REFERENCE BOOKS:

- Day. R A. "Scientific English: A Guide for Scientists and Other Professional". 2nd Edition, Hyderabad: Universities Press, 2000 Print.
- Ashraf Rizvi M. "Effective Technical Communication". McGraw Hill Education, 2017 Print.
- Eastwood J. "Oxford Practice Grammar". Oxford University Press, 2000 Print.
- Swan M and Walter C. "Oxford English Grammar Course Advanced Learner's Book", Oxford University Press, Revised Edition, 2019.
- Dale, Carnegie. "The Quick and Easy Way to Effective Speaking". JAICO Publishing House, 2019 Print.

- N. Krishnaswamy and T. Sri Raman. “Creative English for communication”, 3rd Edition Laxmi Publications Private Limited, 2022.

Course Name: ENVIRONMENT AND PUBLIC HEALTH

Credits:1

Course Description:

This course explores the intricate relationship between the environment and public health, addressing environmental determinants like air, water, and soil quality and their effects on human health. Learners will examine the impacts of climate change, global warming, and industrial practices on public health, including mitigation and adaptation strategies. The study of contemporary diseases includes communicable and non-communicable illnesses, prevention, and control through hygiene, nutrition, and health education. Epidemiological perspectives highlight disease surveillance, occupational health, environmental management policies, and integrative health systems like AYUSH. Emphasis is placed on sustainable practices, community health initiatives, and global responses to public health challenges.

Course Objectives:

- Discuss the environmental determinants of health and climate impacts
- Analyse disease management and epidemiological approaches

Course Outcomes:

At the end of the course, the learner will be able to:

- Analyse environmental factors and climate change impacts to develop strategies for improving public health
- Apply epidemiological methods and health education to prevent diseases and promote sustainable public health practices

Unit 1	Environment And Health: Introduction to Environment and Health, Environmental Determinants of Health, Public Health Framework, Public health systems and environmental health, Ecosystem Services and Health, Effect of Quality of Air, Water, and Soil on Health, Air Quality and Human Health, Water Quality and Health Impacts, Soil Quality and Health Effects, Perspectives on Individual Health: Nutritional, Socio-Cultural, and Developmental Aspects, Nutritional Aspects of Health, Socio-Cultural Determinants of Health, Developmental Aspects of Health, Dietary Diversity for Good Health, Introduction to Dietary Diversity, Balanced Diet and Essential Nutrients, Malnutrition and Dietary Patterns, Sustainable Diets and Public Health, Human Development Indices and Public Health, Overview of Human

	Development Indices (HDI), Health Indicators and Public Health, Social Determinants of Health and Human Development, Global and National Health Strategies
Unit 2	Climate Change: Introduction to Climate Change, Definition and causes of climate change, Global Warming and Its Impact ,Public Health Impacts of Global Warming, Agricultural Practices and Climate Change, Chemical Agriculture and Its Impact on Climate, Industrial Agricultural Practices, Public Health Implications of Agricultural Practices, Industrial Technologies and Climate Impact, Use of Non-Biodegradable Materials, Industrial Pollution and Emissions, Public Health Consequences of Industrial Technologies, Manifestations of Climate Change on Public Health, Burning of Fossil Fuels and Public Health Impacts, Automobile Emissions and Public Health, Acid Rain and Health Effects, Climate Change Mitigation and Public Health Strategies, Climate Change Mitigation Strategies, Public Health Adaptation to Climate Change, Global Health Responses to Climate Change
Unit 3	Diseases In Contemporary Society: Introduction to Health and Diseases, Definition of Health, Need for Good Health, Factors Affecting Health, Types of Diseases, Specific Types of Diseases, Personal Hygiene and Nutrition, Importance of Personal Hygiene, Nutrition and Balanced Diet, Food Habits and Cleanliness, Food Adulterants, Avoiding Harmful Substances, Communicable Diseases, Understanding Communicable Diseases, Epidemic and Endemic Diseases, Management of Hygiene in Public Places, Prevention and Control Strategies, Infectious Diseases, The Role of Sanitation in Infectious Diseases, Poverty and Infectious Diseases, Emerging and Re-emerging Infectious Diseases, Antimicrobial Resistance, Health Promotion and Education, Non-Communicable Diseases (NCDs), Overview of Non-Communicable Diseases, Lifestyle Factors and NCDs, Built Environment and Health, Managing Diabetes and Hypertension, Public Health Approaches to NCDs
Unit 4	Epidemiological Perspectives In Public Health: Introduction to Epidemiology, Definition and scope of epidemiology, Disease Burden and Surveillance, Alternative Systems of Medicine, Universal Immunization Programme (UIP), Reproductive Health and YUVA Programme, Occupational Health and Safety, Importance of occupational health, Physical Health Hazards, Chemical Health Hazards, Biological Health Hazards, Prevention and Control of Occupational Diseases, Environmental Management Policies, Importance of environmental management, Municipal Solid Waste Management, Solid Waste Management in Urban and Rural Areas, Municipal Solid Waste Rules and Regulations, Environmental Protection Legislation, Public Health Interventions, Health Promotion Strategies, Disease Prevention Strategies, Community Health Initiatives, Policy Advocacy for Public Health, Evaluation of Public Health Programs, Integrative Approaches to Public Health, Integrating AYUSH in Public Health, Youth Engagement in Health Issues, Environmental Health and Sustainable Practices, Future Challenges in Public Health

TEXT BOOKS:

- Indian Academy of Paediatrics., Guidebook on Immunisation. mfc bulletin, 45- 50, 2011.
- Nandini N, Sunitha N. and Sucharita Tandon, Environmental Studies, Sapna Book

House, Bangalore, 2007.

- Michel, Mckinney, Robert and Logan, Environmental Science – Systems & Solutions. Jones & Barlett Publishers, Canada, 2007.
- Park, K. Preventive and Social Medicine. Benarsi Das Publications, (pp. 16- 19,24-27), 2011.
- Sadgopal, M., & Sagar, A. Can Public Health open up to the AYUSH Systems and give space for People's views of health and disease?, 2007, July-September.
- Sekhsaria, P. Conservation in India and the Need to Think Beyond 'Tiger vs. Tribal'. Biotropica, 39(5), 575-577, 2007.
- Tyler Miller and Scott E. Spoolman 'Environmental Science', 13th edition First Indian Reprint Chapters 14-17, 2012.
- UNDP. The Human Development Report, The Rise of the South: Human Progress in Diverse World. New York: UNDP, (also available in Hindi), 2013.
- Wani, M., & Kothari, A. Protected areas and human rights India: the impact of the official conservation model on local communities. Policy Matters, 100- 114, 2007, July 15.

DAYANANDA SAGAR UNIVERSITY

Semester: III

Course Name: SOFTWARE ENGINEERING AND TESTING

Credits: 4

Course Description:

This course introduces software engineering principles, starting with the nature of software, process models, and software myths. Learners will explore prescriptive models like Waterfall, Incremental, and Evolutionary approaches, followed by an in-depth study of requirements engineering, elicitation, and modelling using UML diagrams and class-based techniques. Agile methodologies, including Extreme Programming (XP), are introduced to highlight adaptive development practices. The course also covers software testing fundamentals, test case creation, and quality evaluation. Emphasis is placed on advanced testing techniques such as white-box testing, including control flow, basis path, and data flow testing, ensuring a comprehensive understanding of software quality and development processes.

Course Objectives:

- Discuss the software engineering processes and models
- Apply requirements engineering and modelling techniques
- Explore agile development and process models
- Develop and evaluate effective testing strategies
- Implement advanced white-box testing techniques

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain the nature of software, software processes, process models, and key practices in software development.
- Analyse requirements engineering
- Apply agile development principles
- Implement software testing techniques
- Perform advanced white-box testing

Unit 1	Introduction to Software Engineering: The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths, How it all starts
Unit 2	Process Models Overview: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models
Unit 3	Prescriptive Process Models: Waterfall Model, Incremental Process Models, Evolutionary Process Models, Concurrent Models, Specialized Process Models
Unit 4	Introduction to Requirements Engineering: Requirements Engineering, Establishing the Groundwork, Eliciting Requirement, Developing Use Cases
Unit 5	Building the Requirements Model: Building the Requirements Model, Negotiating Requirements, Validating Requirements
Unit 6	Requirements Modeling and Analysis: Requirement Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling
Unit 7	Introduction to Agile Development: What Is Agility?, Agility and the Cost of Change, What Is an Agile Process?
Unit 8	Agile Process Models: Extreme Programming (XP), Other Agile Process Models, A Toolset for Agile Processes
Unit 9	Principles That Guide Practice: Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity
Unit 10	Fundamentals of Testing: Guiding Principles of Testing, Composition of a Testing Team, Essential Skills of a Tester, Software Testing Life Cycle (STLC)
Unit 11	Test Cases and Quality Evaluation: Types of Testing, Test Cases, Evaluating the Quality of Test Cases
Unit 12	Types of Testing: Functional Testing, Non-Functional Testing, Acceptance Testing, Regression Testing
Unit 13	White Box Testing Techniques: Introduction to Control Paths, Control Flow Testing, Basis Path Testing, Loop Testing
Unit 14	Advanced White Box Testing: Data Flow Testing, Slice-Based Testing, Pitfalls of White Box Testing, Tools for White Box Testing

TEXTBOOKS:

- Anirban Basu, Software Quality Assurance, Testing and Metrics, 2015, PHI.
- B.B. Agarwal, M. Gupta, S.P. Tayal, Software Engineering and Testing: An Introduction (Computer Science), Infinity Science Press, April 2009.
- Roger S. Pressman: Software Engineering-A Practitioner's approach, 9th Edition, 2023, Tata McGraw Hill.

DAYANANDA SAGAR UNIVERSITY

Course Name: PYTHON PROGRAMMING

Credits: 3

Course Description:

This course comprehensively introduces Python programming, covering its key features, syntax, and foundational concepts such as variables, data types, operators, and input/output operations. Learners will explore collection data types like strings, lists, tuples, dictionaries, and sets with associated methods. Core control structures and function definitions are examined in detail, including lambda functions, decorators, and generators. Advanced topics include exception handling, modules like Turtle, Date-Time, and Calendar, and data manipulation using NumPy and Pandas. The course introduces regular expressions, file handling, and object-oriented programming concepts, focusing on inheritance, polymorphism, and operator overloading for advanced application development.

Course Objectives:

- Discuss Python basics and data types
- Apply control structures, functions, and exceptions
- Utilise modules, libraries, and data-handling tools
- Implement OOP principles and advanced concepts

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain Python features, data types, operators, and input/output operations
- Implement loops, conditionals, user-defined functions, iterators, generators, and exception handling
- Work with modules, Turtle, NumPy, Pandas, and perform file handling and regular expressions
- Apply classes, objects, inheritance, polymorphism, and operator overloading for advanced programming tasks

Unit 1	Introduction to Python: Features of Python, Comments, Literals, Variables, Identifiers, Operators, Data Types, Input / Output
Unit 2	Collection Data Types: Strings, Lists, Tuples, Dictionaries, Sets, Methods for each data type
Unit 3	Control Structures: If statement, Else statement, Elif statement, While statement, For statement, Break statement, Continue statement, Pass statement
Unit 4	Functions: Defining Functions, Calling Value-Returning Functions, Calling Non-Value-Returning Functions, Parameter Passing, Keyword Arguments, Default Arguments, Variable Scope, Iterators, Generators, Decorators, Lambda Functions
Unit 5	Exception Handling: Errors and Exceptions, Exception Types, Exception Handling using try & except, User-Defined Exceptions
Unit 6	Modules and Libraries: Introduction to Turtle Library, Date and Time Module, Calendar Module
Unit 7	NumPy and Pandas: NumPy Arrays, Computation on NumPy Arrays, Pandas Series and DataFrame, Descriptive Statistics, Handling Missing Data, Data Wrangling with Pandas
Unit 8	Regular Expressions and File Handling: Regular Expressions, Special Symbols and Characters, Text Files, Opening, Reading, and Writing Files
Unit 9	OOP Concepts in Python: OOP Concepts, Classes and Objects, Class Variables
Unit 10	Advanced OOP in Python: Inheritance, Types of Inheritance, Polymorphism, Operator Overloading

TEXTBOOKS:

- Practical Programming: An Introduction to Computer Science Using Python, 3rd edition, Paul Gries, Jennifer Campbell, Jason Montojo, 2017, The Pragmatic Bookshelf.
- Charles Diernach: Introduction to computer science using python: A computational problem-solving, 2015, Wiley.

REFERENCE BOOKS:

- Allen B Downey, "Think Python:How to think like a Computer Scientist", 2nd edition, updated for Python 3, 2016.
- Mark Lutz, "Programming Python", O'Reilly Media, 4th edition, 2011.

Course Name: MATHEMATICS - III

Credits: 4

Course Description:

This course introduces foundational and advanced mathematical concepts essential for computer applications. It begins with matrices, their types, operations, and applications, including determinants and invertible matrices. Learners explore linear programming using graphical methods, vector calculus with scalar and vector functions, and group theory concepts like subgroups and homomorphism. The course advances into ring theory, vector spaces, and linear algebra techniques such as Gauss elimination, LU decomposition, and eigenvalues. Principles of counting, including permutations and combinations, are also covered. Finally, learners learn recurrence relations and their solutions, equipping them with mathematical tools for algorithmic design and analysis.

Course Objectives:

- Analyse the fundamental concepts of information security
- Devise the various scanning tools and apply these methods in real-time applications
- Discuss the enumeration tools and techniques
- Get the idea of system hacking with various tools and real-time applications.
- Illustrate the idea of Penetration testing tools and techniques.

Course Outcomes:

At the end of the course, the learner will be able to:

- Use propositional and predicate logic in knowledge representation and truth verification
- Demonstrate the application of discrete structures in different fields of computer science
- Solve problems using recurrence relations and generating functions
- Different mathematical proof techniques are applied to prove theorems in the courses
- Compare graphs, trees and their applications

Unit 1	Matrices: Introduction, Definition of a matrix, Types of matrices, Operations on Matrices, Transpose of a matrix, Symmetric and skew symmetric matrix, Invertible matrices
Unit 2	Determinants: Introduction, Determinant, Area of a triangle using determinants, Minors and cofactors, Adjoint of a matrix, Applications of matrices and determinants
Unit 3	Linear Programming: Linear inequality, Linear programming, Linear programming problems, Objective function, Graphical method
Unit 4	Vector Calculus: Vector function of a single variable, Scalar and vector point function, Partial differentiation, Gradient of a scalar function, Gradient of Matrices
Unit 5	Group Theory: Introduction, Definition of a group, Properties of Groups, Sub-groups, Cosetns, Homomorphis
Unit 6	Ring Theory: Definition of a matrix, Properties of Rings, Homomorphism and isomorphism, Integral Domains and Fields
Unit 7	Vector Spaces: Definition, Properties, Sub-spaces, Spanning sets, Dependence and Independence
Unit 8	Linear Algebra 1: Norms, Some preliminaries Elementary transformations, Rank of matrix, Gauss elimination method
Unit 9	Linear Algebra 2: Gauss seidel method, Gauss Jordan method, Power method, Quadratics forms
Unit 10	Matrix Decomposition: Determinants and Trace, Eigenvalues and Eigenvectors, LU Decomposition method, Singular value decomposition, Gradient of Matrices
Unit 11	Principles of counting: Sum Rule and Product rule, Permutations, Combinations, Permutation with repetition
Unit 12	Recurrence Relations: Introduction, First order Recurrence Relations, Second order Recurrence Relations.

TEXTBOOKS:

- Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education. 2006.

REFERENCE BOOKS:

- Basavaraj S Anami and Venakanna S Madalli: Discrete Mathematics – A Concept based approach, Universities Press, 2016.
- Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
- Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
- D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications,

Thomson, 2004.

- Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

Course Name: COMPUTER NETWORKS

Credits: 4

Course Description:

This course introduces networking fundamentals, models, and protocols. Topics include the basics of OSI and TCP/IP models, data transmission, routing, switching, and network security. Learners will explore protocols like HTTP, FTP, TCP, and IP while understanding the design and management of local area networks (LANs) and wide area networks (WANs). By the end of the course, learners will be proficient in networking concepts, troubleshooting, and optimising network systems for secure communication.

Course Objectives:

- Discuss the network architecture, protocols, and topologies
- Analyse data transmission and media access techniques
- Explore IP addressing, subnetting, and routing protocols
- Apply transport layer principles for reliable communication
- Examine network applications and application-layer protocols

Course Outcomes:

At the end of the course, the learner will be able to:

- Explore the basics of Computer Networks and the physical layer
- Summarise the data link layer and its protocols
- Explore routing mechanisms and different routing protocols
- Identify the issues of the Transport layer to analyse the congestion control mechanism
- Explain the principles of application layer protocols

Unit 1	Introduction to Networks: Building a Network, Network Edge and Core, Layering and Protocols, OSI Reference Model, Network Topologies, Internet Architecture
Unit 2	Networking Devices and Data Transmission: Networking Devices, Signal Characteristics, Data Transmission, Physical Links and Transmission Media
Unit 3	Signal Encoding and Channel Access Techniques: Signal Encoding Techniques, Channel Access Techniques, TDM, FDM, RPC
Unit 4	Link Layer Fundamentals: Link Layer Services, Framing, Error Control, Flow Control

Unit 5	Media Access Control and Ethernet: Media Access Control, Ethernet, CSMA/CD
Unit 6	LAN Technologies and Wireless LANs: Token Ring, FDDI, Wireless LANs, CSMA/CA
Unit 7	Switching Techniques and Routing Fundamentals: Circuit Switching, Packet Switching, Virtual Circuit Switching, Routing Basics
Unit 8	Internet Protocol and Addressing: IP, Global Address, Datagram Forwarding, Subnetting, CIDR
Unit 9	Network Layer Protocols: ARP, DHCP, ICMP, IPv6
Unit 10	Routing Protocols: RIP, OSPF, BGP
Unit 11	Transport Layer Overview and UDP: Overview of Transport Layer, UDP
Unit 12	TCP and Reliable Data Transfer: TCP, Reliable Byte Stream, Connection Management, Flow Control, Retransmission
Unit 13	Congestion Control and Avoidance: Congestion Control, Congestion Avoidance
Unit 14	Network Applications and Protocols: Principles of Network Applications, WEB and HTTP, FTP, E-MAIL (SMTP, POP3), TELNET, DNS, SNMP

TEXTBOOKS:

- Behrouz A. Forouzan, Data Communications and Networking, 6th Edition, Tata McGraw Hill, 2022.
- Computer Networks - Andrew S Tanenbaum, 6th Edition, 2022, Pearson Education.

REFERENCE BOOKS:

- James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, 7th Edition, Pearson Education, 2017.
- Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, 6th Edition, Morgan Kaufmann Publishers Inc., 2020.
- William Stallings, “Data and Computer Communications”, 10th Edition, Pearson Education, 2017.
- Larry L. Peterson and Bruce S. Davie,”Computer Networks: A systems approach”,Morgan Kaufmann Publishers, USA, 6th Edition, 2020.
- James F. Kurose, Keith W. Ross,”Computer Networking – A Top-Down Approach Featuring the Internet”, Pearson Education, New Delhi, 7th Edition, 2016.
- Prakash C Gupta,” Data Communication and Computer Networks”, Prentice Hall of India, New Delhi, 2nd Edition, 2014.
- Ying-Dar Lin, Ren-Hung Hwang and Fred Baker, “Computer Networks An Open

Source Approach “,McGraw Hill Publisher, USA, 2011.

- Andrew S Tanenbaum, David J. Wetherill “Computer Networks”, Prentice Hall of India/ Pearson Education, New Delhi, 2010.

DAYANANDA SAGAR UNIVERSITY

Course Name: ANALYSIS AND DESIGN OF ALGORITHMS

Credits: 4

Course Descriptions:

This course covers the core principles and methods of algorithm design and analysis, essential for solving computational problems efficiently. It begins with the fundamentals of algorithms, their properties, and complexity analysis using asymptotic notations like Big O, Omega, and Theta. Techniques for solving recurrence relations, including the master theorem, are introduced. Key strategies like divide-and-conquer, greedy methods, dynamic programming, backtracking, and branch-and-bound are explored with practical applications, such as sorting, graph problems, and NP-complete challenges. Topics include the Knapsack Problem, the Traveling Salesperson Problem, and optimal solutions for binary search trees, preparing learners for advanced computational problem-solving.

Course Objectives:

- Discuss the algorithm fundamentals and complexity analysis
- Apply recurrence relations and divide-and-conquer techniques
- Utilise greedy and dynamic programming methods effectively
- Solve problems using backtracking and branch-and-bound techniques
- Analyse NP-Hard, NP-Complete problems and optimisation strategies

Course Outcomes:

At the end of the course, the learner will be able to:

- Interpret the various methods for the analysis of algorithms
- Discuss the designing algorithms for various computing problems
- Explain the time and space complexity of algorithms
- Critically analyse the different algorithm design techniques for a given problem
- Explore and modify existing algorithms to improve efficiency

Unit 1	Fundamentals of Algorithms: Definition of Algorithms, Properties of Algorithms, Recurrence and Non-Recurrence Algorithms, What is an algorithm?, Input and output specifications, Definiteness, effectiveness, finiteness, Examples of Recurrence and Non-Recurrence Algorithms
Unit 2	Analysis of Algorithms: Asymptotic Notations, Time and space complexity, Big O notation, Omega and Theta notations, Significance of asymptotic analysis
Unit 3	Solving Recurrence Relations: Substitution method, Recursion tree method, Master theorem, Examples and practice problems
Unit 4	Complexity Analysis of Algorithms: Insertion Sort, Radix Sort, Linear Search, Finding Factorial, Binary Search, Step-by-step complexity calculation, Best, average, and worst-case scenarios, Comparison of algorithm efficiencies, Practical implications of complexity analysis
Unit 5	Introduction to NP-Hard and NP-Complete Problems: NP-Hard Problems, NP-Complete Problems vs. NP problem, Definitions and differences between NP, NP-Hard, and NP-Complete, Examples of NP-Complete problems, Significance in computational theory
Unit 6	Divide-and-Conquer Method: The General Method, Finding Maximum and Minimum Elements, Quick Sort, Merge Sort, Matrix Multiplication, Concept of dividing problems into subproblems, Algorithm for finding max and min, Quick Sort algorithm and analysis, Merge Sort algorithm and analysis, Strassen's Matrix Multiplication method
Unit 7	Greedy Method Fundamentals: The General Method, Optimal Storage on Tapes, Knapsack Problem, Principles of greedy algorithms, Greedy choice property, Problem of optimal storage on tapes, Greedy solution to the Knapsack problem
Unit 8	Greedy Method Applications: Minimum Spanning Trees, Single Source Shortest Path, Floyd-Warshall's Algorithm, Prim's and Kruskal's algorithms for MST, Dijkstra's algorithm for shortest paths, Application of greedy method in graph algorithms, Understanding Floyd-Warshall's algorithm for all pairs shortest paths
Unit 9	Dynamic Programming Concepts: The General Method, - All Pairs Shortest Path, Principle of optimality, Overlapping subproblems, Memoization vs. tabulation, Implementation of the All Pairs Shortest Path problem
Unit 10	Dynamic Programming Applications: Optimal Binary Search Tree - Multistage Graphs, Constructing an optimal BST, Calculating search costs, Solving multistage graph problems, Cost optimisation in multistage decision processes
Unit 11	Backtracking Concepts: The General Method - Solution Space and Tree Organization, Constructing solution space trees, Depth-first search strategies, Pruning and bounding techniques, Framework for backtracking algorithms
Unit 12	Backtracking Applications: The Eight Queens Problem, Sum of Subsets Problem, Graph Coloring, Knapsack Problem, Formulating the Eight Queens problem, Recursive solutions, Coloring graphs with minimal colors, Backtracking solution to the Knapsack problem, Analysing time complexity of backtracking algorithms
Unit 13	Branch and Bound Concepts: The General Method, LC Search, LC Branch and Bound, FIFO Branch and Bound, Difference between backtracking and branch and bound, Least Cost (LC) search strategies, Implementing LC branch and bound, FIFO branch and bound approaches

Unit 14	Branch and Bound Applications: 0/1 Knapsack Problem, Traveling Salesperson Problem, Efficiency Considerations, Applying branch and bound to the Knapsack problem, Solving the Traveling Salesperson Problem (TSP), Analysing efficiency and computational complexity, Strategies to improve performance of branch and bound algorithms

TEXTBOOKS:

- Thomas H.Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", MIT Press, England, 4th Edition, 2022.

REFERENCE BOOKS:

- Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Computer Algorithms", Computer Science Press, New York, 1998.
- Anany V Levitin, "The Design & Analysis of Algorithms", 3rd Edition, Pearson India, 2014.
- Alfred V. Aho, John E. Hopcroft and Jeffrey D. U II man, "The Design and Analysis of Computer Algorithms", Addison-Wesley Publishing Company, 1974.

Course Name: OPERATING SYSTEMS

Credits: 3

Course Description:

This course covers the principles and functionalities of modern operating systems. Topics include process management, memory management, file systems, I/O management, and synchronisation. Learners will learn concepts like deadlocks, scheduling algorithms, and virtual memory. Emphasis is placed on understanding OS internals and the role of system software. By the end, learners will be equipped to analyse, evaluate, and optimise operating systems for better performance and resource management.

Course Objectives:

- Outline the different types of Operating Systems and their services
- Discuss the different process scheduling algorithms and synchronisation techniques
- Explain the virtual memory concepts
- Describe the concepts of secondary memory management

Course Outcomes:

At the end of the course, the learner will be able to:

- Design various Scheduling algorithms
- Explain deadlock, prevention and avoidance algorithms
- Compare and contrast multiple memory management schemes
- Discuss the storage management and file structures

Unit 1	Introduction to Operating Systems: Operating System as an Extended Machine, Operating System as a Resource Manager, History of Operating Systems, Basic OS Concepts, Definitions and Functions of Operating Systems, OS as an Interface between Hardware and Users, Evolution of Operating Systems (First to Present Generations), Basic Concepts: Processes, Files, Shell
Unit 2	Operating System Structures: Operating System Services- User Operating System Interface, System Calls and Types, OS Design and Implementation, Virtual Machines, Debugging, System Boot, Types of OS Services (Program Execution, I/O Operations, File-System Manipulation, Communications, Error Detection), Command-Line Interface (CLI) and Graphical User Interface (GUI), Types of

	System Calls (Process Control, File Management, Device Management, Information Maintenance, Communications), Design Goals and Approaches, Concept of Virtual Machines and Their Benefits, Debugging Techniques and Tools, Bootstrapping Process
Unit 3	Process Management: Process Model, Process Creation and Termination, Process Hierarchies, Process States, Process Implementation, Definition of a Process, Process Lifecycle and Control Blocks, Parent and Child Processes, Process States (New, Ready, Running, Waiting, Terminated), Context Switching and Implementation Details
Unit 4	Process Synchronisation: Race Conditions, Critical Sections, Mutual Exclusion, Peterson's Solution, Synchronization Hardware, Semaphores, Mutexes, Monitors, Message Passing, Atomic Transactions,, Issues with Concurrent Processes Critical Section Problem and Requirements, Software Solutions to Mutual Exclusion (Peterson's Algorithm), Hardware Support for Synchronization (Test-and-Set, Swap), Semaphore Operations and Usage, Mutex Locks and Their Applications, Monitor Constructs for Synchronisation, Inter process Communication via Message Passing, Ensuring Atomicity in Transactions
Unit 5	Threads and Concurrency: Multithreading Models, Thread Libraries, Threading Issues, Threading in Java, Classic Synchronization Problems, User-Level vs. Kernel-Level Threads, Multithreading Models (Many-to-One, One-to-One, Many-to-Many), POSIX Threads, Windows Threads, Java Threads, Thread Cancellation, Signal Handling, Thread Pools, Implementation of Threads in Java, Dining Philosophers Problem, Readers-Writers Problem, Programming Solutions to Synchronization Problems
Unit 6	CPU Scheduling: Scheduling in Batch Systems, Scheduling in Interactive Systems, Scheduling in Real-Time Systems, Scheduling Criteria, Scheduling Algorithms, Thread and Multiprocessor Scheduling - Algorithm Evaluation, CPU Scheduling Objectives and Criteria (CPU Utilisation, Throughput, Turnaround Time, Waiting Time, Response Time), Scheduling Algorithms: First-Come-First-Served (FCFS), Shortest Job First (SJF), Priority Scheduling, Round Robin (RR), Multilevel Queue, Multilevel Feedback Queue, Thread Scheduling Issues, Multiprocessor Scheduling Techniques, Comparative Evaluation of Scheduling Algorithms
Unit 7	Deadlocks: Resources, Principles of Deadlock, Methods for Handling Deadlocks, Ostrich Algorithm, Banker's Algorithm, Detection and Recovery, Deadlock Prevention, Deadlock Avoidance, Types of Resources (Reusable, Consumable), Necessary Conditions for Deadlock (Mutual Exclusion, Hold and Wait, No Preemption, Circular Wait), Ignoring Deadlocks (Ostrich Algorithm), Deadlock Detection Algorithms, Recovery Strategies (Process Termination, Resource Preemption), Deadlock Prevention Techniques, Banker's Algorithm for Deadlock Avoidance
Unit 8	Memory Management: Main Memory Management, Swapping, Contiguous Memory Allocation, Paging, Structure of Page Table, Segmentation, Examples, Memory Allocation Strategies (Fixed Partitioning, Dynamic Partitioning), Swapping Processes In and Out of Memory, Contiguous Allocation Methods, Paging Concepts and Page Table Structures (Hierarchical Paging, Hashed Page Tables, Inverted Page

	Tables), Segmentation and its Implementation, Combining Paging and Segmentation, Practical Examples of Memory Allocation
Unit 9	Virtual Memory and Advanced Memory Management: Virtual Memory, Demand Paging, Copy-on-Write, Page Replacement Algorithms, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory, Benefits of Virtual Memory Systems, Implementation of Demand Paging, Copy-on-Write Optimization, Page Replacement Algorithms (FIFO, LRU, Optimal, Second Chance), Frame Allocation Strategies (Equal, Proportional, Priority-Based), Causes and Solutions for Thrashing, Use of Memory-Mapped Files for File I/O, Kernel Memory Allocation Techniques (Buddy System, Slab Allocator), Memory Management Utilities
Unit 10	Virtualisation: Requirements for Virtualization, Type 1 Hypervisors, Type 2 Hypervisors, Paravirtualization, Memory Virtualization, I/O Virtualization, Virtual Appliances, Virtual Machines on Multicore CPUs, Licensing Issues, Case Study: Windows, Utilities for Monitoring and Managing Memory, Fundamental Concepts of Virtualization, Role and Functionality of Hypervisors, Comparison between Type 1 (Bare Metal) and Type 2 (Hosted) Hypervisors, Principles of Paravirtualization, Techniques for Virtualizing Memory and I/O Devices, Deployment of Virtual Appliances, Managing Virtual Machines on Multicore Processors, Legal and Licensing Considerations in Virtualization, Application of Virtualization in Windows Operating System

TEXTBOOKS:

- Silberschatz A, Galvin P and Gagne G “Operating Systems Concepts”, John Wiley & Sons, USA, 9th Edition, Wiley.
- Andrew S Tanenbaum, “Operating Systems Design and Implementation”, Prentice Hall of India, New Delhi, 3rd Edition, 2015.

REFERENCE BOOKS:

- Deitel H M, Deitel J P, David R Choffnes, " Operating Systems", Prentice Hall of India, New Delhi, 2004.
- Andrew S. Tanenbaum, “Modern Operating Systems”, Prentice Hall of India, New Delhi, 5th Edition, 2024.
- Mark Russinovich, David A. Solomon, Alex Ionesco, ”Windows Internals: Including Windows Server 2008 and Windows Vista”, Microsoft Press, Cambridge, England, 2009.
- Gary Nutt, “Operating Systems”, Addison Wesley, USA, 2004.

DAYANANDA SAGAR UNIVERSITY

Semester: IV

Course Name: OBJECT-ORIENTED ANALYSIS AND DESIGN

Credits: 4

Course Description:

This course introduces object-oriented analysis and design principles and methodologies, focusing on building structured, efficient systems. It explores system complexity, the evolution of the object model, and relationships among objects and classes. Topics include creating quality classes, object-oriented methodologies (Rumbaugh, Booch, Jacobson), and the Unified Modelling Language (UML). Learners will learn to develop structure and behavioural diagrams, use UML tools, and perform object-oriented analysis, including use-case modelling. Practical case studies such as the ATM and Online Shopping System provide hands-on experience, equipping learners with the skills to model real-world systems effectively.

Course Objectives:

- Analyse the non-recursive and recursive algorithms.
- Devise the Brute Force and Divide and Conquer techniques
- Explain the Decrease and Conquer, Transform and Conquer algorithm design techniques, and Time versus Space Trade-offs
- Outline the idea of the greedy and dynamic programming methods
- Illustrate the concept of Backtracking and Branch and Bound algorithm design techniques

Course Outcomes:

At the end of the course, the learner will be able to:

- Articulate the fundamental concepts and object model and analyse the system's complexity
- Identify the classes, objects and their relationships in the object-oriented model
- Explain object-oriented methodologies, pattern frameworks, and a unified approach
- Discuss the concept of UML Design and its types of diagrams

- Apply appropriate technologies, algorithms and approaches for real-time issues

Unit 1	Complexity of Systems: Complexity and structure of complex systems, Inherent complexity of software, Attributes of complex systems
Unit 2	Object Model Evolution: Foundations of Object model, Elements of Object model: Major (Abstraction, Encapsulation, Modularity, Hierarchy), Minor (Typing, Concurrency, Persistence)
Unit 3	Nature and Relationships: Nature of an object, Relationships among objects, Nature of classes, Relationships among classes
Unit 4	Building Quality Classes: Interplay of classes and objects, Building quality classes and objects, Classification: Importance and Identifying classes and objects
Unit 5	Object-Oriented Methodologies I: Introduction to OOP methodologies, Survey of methodologies (Rumbaugh, Booch), real-time examples
Unit 6	Object-Oriented Methodologies II: Detailed overview of methodologies: Jacobson, Patterns, Frameworks, Unified Approach
Unit 7	Introduction to UML: Purpose and overview of UML, Why UML is important, Interaction, Significance, Advantages and Disadvantages
Unit 8	UML Diagrams: Types of UML diagrams, Structure Diagrams and its types, Examples
Unit 9	Behavioral Diagrams: UML Behavioral diagrams and overview of different behavioral diagram types, real-time examples
Unit 10	UML Tools: Tools for UML design- Star UML, LucidChart, case studies
Unit 11	Introduction to OO Analysis: Introduction and difficulties in object-oriented analysis, Why analysis is a challenging activity
Unit 12	Use-case Modeling: Creating and using use-case models, Developing effective documentation
Unit 13	Case Study: ATM System- Practical application of concepts: Analysis and modeling of ATM system
Unit 14	Case Study: Online Shopping System- Practical application of concepts: Analysis and modeling of Online Shopping System

TEXTBOOKS:

- Grady Booch, Robert A.Maksimchuk, Michael. W. Engle, Bobbi J. Young, JIM Conallen, Kelli A. Houston “Object Oriented Analysis and Design with Applications”, Pearson Education Inc., USA, 2010.
- Ali Bahrami, “Object Oriented System Development”, McGraw Hill International Edition, Singapore, 2008.

REFERENCE BOOKS:

- Rumbaugh J, Blaha M, Premerlani W, Eddy F and Lorensen W, “Object Oriented Modeling and Design”, Prentice Hall of India/ Pearson Education, New Delhi, 2004.
- Kendall Scott, martin Fowler, “UML Distiled : A brief guide to the standard Object modeling Language “, Addison Wesley, USA, 2009.
- Atul Kahate, “ Object Oriented Analysis and Design “, Tata McGraw-Hill , New Delhi 2007. 4. Sudha Sadasivam G., “ Object-Oriented Analysis and Design”, Macmillan India, New Delhi, 2009.

Course Name: WEB TECHNOLOGIES

Credits: 3

Course Descriptions:

This course focuses on advanced JavaScript and full-stack web development concepts, equipping learners with skills for modern web applications. Learners begin with DOM manipulation, event handling, transitions, animations, and JSON with Fetch API for client-side operations. The course covers asynchronous programming, including callbacks, promises, and async/await, alongside Node.js basics like modules, routing, and RESTful API development. MongoDB operations for database interactions, such as JWT-based authentication and WebSockets for building interactive applications, are introduced. The course culminates in developing and deploying a full-stack application, providing hands-on experience in web development and deployment strategies.

Course Objectives:

- Discuss the JavaScript basics and DOM manipulation
- Implement animations, APIs, and asynchronous programming
- Develop backend services using Node.js and MongoDB
- Apply authentication and real-time communication techniques
- Build and deploy full-stack web applications

Course Outcomes:

At the end of the course, the learner will be able to:

- Make use of javascript to manipulate DOM
- Make use of client-side api like fetch, storage, geolocation, history
- Apply asynchronous programming with nodejs
- Build applications with server-side data fetching, authentication and chatbots with WebSockets
- Interpret the case study of a full-stack application based on nodejs

Unit 1	Basic JavaScript and DOM: Review of basic JavaScript, DOM manipulation, Event handling
Unit 2	Transitions and Animations: Using JavaScript for transitions, animations, and canvas
Unit 3	JSON and Fetch API: Introduction to JSON, Fetch API (HTTP protocol), Client-side storage, examples and case studies
Unit 4	Advanced APIs and Asynchronous Programming: Geolocation, History APIs, Asynchronous programming with callbacks
Unit 5	Node.js Basics: Promises, async/await, Installation of Node.js and MongoDB
Unit 6	Node.js Modules and Routing: Node.js modules, npm, Routing, RESTful API development
Unit 7	MongoDB Operations: MongoDB operations: find, update, insert, delete using JavaScript
Unit 8	Authentication and Websockets: JWT Authentication, Building chatbots using Websockets
Unit 9	Full-Stack Application: Development of a full-stack application (e.g., Google Keep clone)
Unit 10	Deployment: Deployment strategies for web applications, Process of deployment, Significance, Advantages and Disadvantages

TEXTBOOKS:

- Programming JavaScript Applications Robust Web Architecture with Node, HTML5, and Modern JS Libraries by Eric Elliot, 2014.
- Homepage https://drmithileshkumar.com/courses/internet
- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference

REFERENCE BOOKS:

- Robin Nixon, “Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5”,4thEdition, O“Reilly Publications, 2015 (ISBN:978-9352130153).
- Luke Welling, Laura Thomson, “PHP and MySQL Web Development”, 5th Edition, Pearson Education, 2016 (ISBN:978-9332582736).
- Nicholas C Zakas, “Professional JavaScript for Web Developers”, 3rd Edition, Wrox/Wiley India, 2012 (ISBN:978-8126535088).

- David Sawyer Mcfarland, “JavaScript & jQuery: The Missing Manual”, 1st Edition, O‟Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014.

DAYANANDA SAGAR UNIVERSITY

Course Name: DATABASE SYSTEMS

Credits: 3

Course Description:

This course introduces database concepts, relational models, and Structured Query Language (SQL). Learners will explore database design techniques, normalisation, and transaction management. Key topics include indexing, triggers, stored procedures, and data integrity. By the end of the course, learners will have practical experience in designing, implementing, and managing relational databases, ensuring efficient data storage and retrieval. Hands-on projects equip learners to solve real-world database problems.

Course Objectives:

- Discuss database concepts, architecture, and models.
- Design E-R diagrams, enforce relational integrity, and normalise databases
- Develop and optimise SQL queries using indexing
- Explore NoSQL databases, focusing on MongoDB fundamentals

Course Outcomes:

At the end of the course, the learner will be able to:

- Design and implement databases using E-R diagrams, relational models, and integrity constraints
- Develop SQL queries for database creation, retrieval, and manipulation
- Apply normalisation techniques to manage functional dependencies and improve database design
- Optimise database performance using indexing, hashing, and B-Tree techniques

Unit 1	Database Basics: Data, Database, Database management system, Characteristics of database approach, Role of DB administrators and designers
Unit 2	DBMS Architecture and Models: Advantages and disadvantages of DBMS, Data Models, DBMS architecture (Three schema), Data independence, DBMS classifications
Unit 3	Entity-Relationship Model Basics: High-level conceptual data models, Entity types, Entity sets, Attributes, Keys
Unit 4	E-R Diagrams: Relationships, Relationship types, Roles, Structural constraints, Weak entities, Drawing E-R diagrams
Unit 5	Integrity and Operations: Relations, Integrity constraints (domain, entity, referential), Basic Relational Algebra operations
Unit 6	Normalisation and Functional Dependencies: Functional dependencies, Normalisation (1NF, 2NF, 3NF, BCNF), real-time examples and case studies
Unit 7	SQL Basics: SQL Queries, subqueries, correlated subqueries, views, updating databases through views, Update, Delete operations
Unit 8	Indexing and Hashing: Single-level ordered indexes (primary, clustering, secondary), Multi-level indexes, B-trees, Hashing concepts
Unit 9	NoSQL Fundamentals: NoSQL databases, NoSQL database features, Types of NoSQL databases, RDBMS vs. NoSQL
Unit 10	MongoDB Basics: MongoDB NoSQL server, Querying in MongoDB Atlas, Architecture of MongoDB, Working of MongoDB

TEXTBOOKS:

- Ramez Elmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Seventh Edition, Pearson/Admission Wesley, 2016.
- Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Seventh Edition, Tata McGraw Hill, 2021.

REFERENCE BOOKS:

- O`neil Patric & O`neil Elizabeth, Database Principles, Programming and Performance, 2nd Edition, Morgan Kaufmann Publishers Inc, 2000.
- Bipin Desai, “An Introduction to Database Systems”, Galgotia Publications, 1991.

Course Name: DATA SCIENCE WITH PYTHON

Credits: 4

Course Description:

This course introduces the foundational concepts of data science, focusing on Python as a core programming tool. It starts with Python basics, including installation, syntax, and core data structures like lists, tuples, sets, and dictionaries. Learners explore essential libraries such as NumPy for array operations, Pandas for data manipulation, and Matplotlib and Seaborn for data visualisation. Key topics include handling missing data, data transformation, and descriptive statistics. The course also delves into hypothesis testing, inferential statistics, and machine learning concepts, covering supervised (linear regression, classification) and unsupervised techniques (clustering, PCA). By the end, learners gain hands-on experience in Python-driven data science workflows.

Course Objectives:

- Discuss the basics of data science and the role Python plays in data analysis
- Learn to manipulate and clean data using core Python data structures and libraries like NumPy and Pandas
- Interpret data visualisation techniques to present data insights using Matplotlib and Seaborn
- Explore the fundamental statistical analyses and hypothesis testing on datasets
- Demonstrate foundational machine learning models using Scikit-learn

Course Outcomes:

At the end of the course, the learner will be able to:

- Apply Python programming principles for data science applications
- Implement core Python data structures for data manipulation
- Utilise Python libraries for numerical and data analysis
- Interpret data visualisations using Matplotlib and Seaborn
- Apply statistical and machine learning techniques to solve real-world problems

Unit 1	Overview of Data Science and Role of Python: Data Science Basics, Python Relevance in Data Science, Overview of Data Science, History of Python, Applications of Python in Data Science
Unit 2	Setting Up Python Environment & Basic Syntax: Installing Python, IDEs, Basic Syntax, Python installation, Jupyter Notebook, Variables, Data Types, I/O
Unit 3	Lists, Tuples, Sets, and Dictionaries: Core Data Structures in Python: Creation, Manipulation, Applications, Pros and Cons
Unit 4	NumPy: Arrays and Operations: Introduction to NumPy, Array Operations: Creation, Indexing, Slicing, Mathematical Operations
Unit 5	Pandas: DataFrames and Data Manipulation: DataFrames, Series, Data Cleaning: Creation, Filtering, Aggregation, Handling Missing Data
Unit 6	Handling Missing Data: Identifying and Filling Missing Data: Drop, Fill Methods, Imputation Techniques
Unit 7	Data Transformation Techniques: Data Scaling and Encoding: Standardization, Normalization, Encoding Categorical Variables
Unit 8	Matplotlib Basics: Creating Plots, Customizing Graphs: Line Plot, Bar Chart, Histogram, Customization
Unit 9	Introduction to Seaborn for Data Visualization: Data Visualization with Seaborn: Pairplot, Heatmap, Advanced Customizations
Unit 10	Descriptive Statistics and Data Distribution: Mean, Median, Mode, Variance, Distribution Plots, Measures of Central Tendency, Measures of Spread
Unit 11	Hypothesis Testing and Inferential Statistics: Statistical Testing Concepts: T-Test, Chi-Square Test, P-Values, Confidence Intervals
Unit 12	Overview of Machine Learning Concepts: Supervised and Unsupervised Learning, Types of ML, Applications, Algorithms Overview
Unit 13	Supervised Learning with Scikit-learn: Linear Regression, Classification Models, Regression Models, Accuracy Metrics, Fitting Models
Unit 14	Unsupervised Learning Techniques: Clustering, Dimensionality Reduction, K-Means, PCA

TEXTBOOKS:

- McKinney, W., Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython (3rd ed.). O'Reilly Media, 2022.
- Raschka, S., & Mirjalili, V. , Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow 2 (3rd ed.). Packt Publishing, 2019. VanderPlas, J., Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly Media., ISBN: 978-1491912058, 2016.

Course Name: LINUX OPEN SOURCE SYSTEM

Credits: 4

Course Description:

This course provides a comprehensive understanding of the Linux operating system, covering its architecture, core distributions, and open-source features. Learners will learn about the Linux file system, process concepts, and essential shell functionalities. Key topics include file handling, permissions, navigation commands, and utilities like grep and chmod. Shell scripting basics and advanced scripting techniques are explored with practical examples. The course introduces system administration, focusing on user, disk, and task management. Additionally, learners will understand network configuration, TCP/IP, DHCP, and security measures such as firewalls and intrusion detection tools. By the end, learners gain hands-on Linux system management skills.

Course Objectives:

- Discuss a versatile operating system
- Use Basic Linux commands and utilities
- Develop a Shell program to solve different problems
- Analyse the Linux Operating System
- Implement Linux Network Services

Course Outcomes:

At the end of the course, the learner will be able to:

- Describe the basics of the Linux Operating System
- Use Basic Linux commands and utilities
- Develop a Shell program for solving different problems
- Maintain Linux Operating System
- Analyse and use Linux Network Services

Unit 1	Overview of Linux OS: Features of Open Source OS, Core Linux Distributions, Architecture, OS Services, System Calls, Run Levels
Unit 2	Linux File System: Hierarchical File System, File System features, Data Structures, Structure of file system, Architecture
Unit 3	Process concepts: Context of process, Context Switch, Process State, State Transition diagram
Unit 4	Shell and Login: Login process, Concept of Shell, Various Linux Shell and their Features, examples and real-time
Unit 5	Basic Commands and Utilities: Locating commands, Internal & External commands, Arguments, Options & Filenames, Online help
Unit 6	Navigating File System: Files, Directories, Paths, Home Directory, Parent-Child Relationship, pwd, cd, mkdir, rmdir, ls
Unit 7	File Handling and Permissions: File Attributes, chmod Command, grep Family: Regular expressions, grep, egrep, fgrep, tr
Unit 8	Vi Editor and Shell Basics: Modes of vi, Shell, sh Command, Pattern Matching, Escaping, Quoting, Redirection, Pipes, Tees
Unit 9	Shell Scripting Basics: Shell Variables, Shell Scripts, read Statement, Command Line Arguments, Positional Parameters
Unit 10	Advanced Shell Programming: Logical Operators, if and case Statements, expr Statement, while, until, for Statements, Sample Scripts
Unit 11	System Administration Overview: Role of Administrator, root Login, su Command, Task Scheduling with cron, Security Maintenance
Unit 12	User and Disk Management: User management (User configuration, password files, managing users/groups), Disk Space management (df, du commands)
Unit 13	Network Configuration: TCP/IP Network address, TCP/IP Configuration files, Network Interfaces, Routes, DHCP Client/Server
Unit 14	Network Security and Firewalls: NFS, Firewall and Internet Security, Network Intrusion Detection, Host-based Intrusion Detection Software (Tripwire)

TEXTBOOKS:

- Linux Command Line and Shell Scripting Bible Edition Blilln, Richard, Bresnahan, Christine Wiley Publication, New Delhi, 4th Edition, 2021.

- Unix Concept and Programming,Das, Sumitabha ,McGraw Hill education, New Delhi, 4th Edition, 2017, ISBN: 978-0070635463.
- Linux Administration A Beginner's Guide,Wale Soyinka, McGraw Hill education, 7th Edition, 2016.

REFERENCE BOOKS:

- LINUX in a NUTTSHELL ,A Desktop quick reference, O'reilly, 6th Edition.

Course Name: DEVOPS – I

Credits: 4

Course Description:

This course provides a foundational understanding of DevOps principles, culture, and practices, highlighting its benefits over traditional approaches. Learners will explore DevOps culture, communication and collaboration tools, and the Infrastructure as Code (IaC) concept, focusing on tools like Terraform and Ansible. Topics include configuration management basics, CI/CD pipelines, and their implementation using Jenkins and GitLab CI/CD. Real-world practices and challenges of DevOps are examined through case studies. Additionally, the course discusses emerging trends and the future of DevOps. By the end, learners will gain practical insights into improving collaboration, automation, and deployment efficiency.

Course Objectives:

- Discuss the DevOps principles, culture, and benefits
- Explore Infrastructure as Code (IaC) and configuration management tools
- Implement CI/CD pipelines using Jenkins and GitLab
- Apply DevOps practices to real-world scenarios
- Analyse DevOps trends and future advancements

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the core concepts of DevOps and its benefits for software development
- Gain exposure to various DevOps methodologies and tools
- Learn about infrastructure as code (IaC) and configuration management
- Implement continuous integration and continuous delivery (CI/CD) pipelines
- Grasp the significance of automation and monitoring in DevOps practices

Unit 1	Introduction to DevOps: What is DevOps? DevOps principles and philosophy, Benefits of adopting DevOps
Unit 2	Traditional vs. DevOps Approach: Differences between traditional and DevOps approaches, Examples of traditional and DevOps Approach
Unit 3	DevOps Culture: Breaking down silos between Dev and Ops teams, Fostering communication and collaboration
Unit 4	Communication Tools: DevOps tools for communication and collaboration, Purpose and significance, examples
Unit 5	Introduction to IaC: Concept of Infrastructure as Code (IaC), Benefits of IaC for infrastructure provisioning
Unit 6	IaC Tools Overview: Introduction to popular IaC tools like Terraform, Ansible, Purpose, Significance
Unit 7	Configuration Management Basics: Configuration management fundamentals, Benefits of configuration management tools
Unit 8	Popular Tools Overview: Introduction to popular configuration management tools (Ansible, Puppet, Chef)
Unit 9	CI/CD Basics: CI/CD pipeline concepts and stages, Examples and case studies
Unit 10	CI/CD Process: Benefits of CI/CD for faster deployments, Process of CI/ CD, General architecture
Unit 11	CI/CD with Jenkins: Implementing CI/CD pipelines with Jenkins, Process, General steps
Unit 12	CI/CD with GitLab CI/CD: Implementing CI/CD pipelines with GitLab CI/CD, examples and case studies
Unit 13	DevOps Real-World Practices: Applying DevOps practices to real-world scenarios, common challenges and solutions
Unit 14	DevOps Trends and Future: Current trends and future direction of DevOps, Real-time case studies, examples

TEXTBOOKS:

- The DevOps Handbook: Gene Kim, Jez Humble, Patrick Debois, 2016.

REFERENCE BOOKS:

- The Phoenix Project, Gene Kim, Kevin Behr, and George Spafford, 2024.
- DevOps on the Microsoft Stack, Wouter de Kort, 2016.

Course Name: IoT

Credits: 4

Course Description:

This course introduces the fundamental concepts, design, and applications of IoT. Learners will learn about IoT characteristics, protocols, communication models, and enabling technologies like cloud computing and wireless sensor networks. Key topics include M2M communication, IoT design requirements, and sensor and actuator technologies. Practical components focus on prototyping IoT systems using Arduino, Raspberry Pi, and sensor-based applications. The course also explores cloud storage models, IoT data analytics, and real-world design challenges. Learners will examine innovative applications, including home automation, industrial automation, and smart agriculture while understanding IoT system integration with modern technologies.

Course Objectives:

- Discuss the basic concepts of the Internet of Things
- Analyse the concept of M2M (machine to machine) with necessary protocols
- Outline the purpose of IoT protocols and apply them in real-time applications
- Design the Software for IoT Applications using Arduino IDE and Raspberry Pi
- Apply the role of big data, cloud computing and data analytics in a typical IoT system

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the design, characteristics and technologies of IoT
- Appraise the role of IoT protocols for efficient network communication
- Illustrate different sensor technologies for sensing real-world entities and identify the applications of IoT in the Industry.
- Apply IoT to develop real-life applications using Arduino programming
- Elaborate on the need for Data Analytics and security in IoT

Unit 1	Introduction to IoT Basics: Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of IoT
Unit 2	IoT Protocols and Models: IoT Protocols, IoT Communication Models and APIs, IoT Enabled Technologies (Wireless Sensor Networks, Cloud Computing, Embedded Systems)
Unit 3	IoT System Management: General Architecture, process steps, IoT System Management using NETCONF-YANG
Unit 4	IoT Design and Applications: Design of IoT, IoT Application Areas, IoT Examples, Layered Architecture of IoT
Unit 5	IoT Communication Protocols: Protocols for IoT (IEEE 802.15.4, Zigbee Architecture, WiFi, LowPAN, LoRaWAN, CoAP)
Unit 6	M2M Communication and IoT: Machine-to-Machine communication, Differences and Similarities between M2M and IoT
Unit 7	IoT Design Requirements: Specification - Requirement, Process, Model, Service, Functional and Operational View
Unit 8	Sensor and Actuator Technologies: Sensors, Actuators, Participatory Sensing, Communication Protocols, RFID Technology, Wireless Sensor Network Technology
Unit 9	Prototyping with Arduino: Introduction to Arduino, Raspberry Pi, Design using Tinkercad, Arduino Programming (Blink LED, Traffic Light Control)
Unit 10	Sensor-based Applications: Creating applications using sensors (Ultrasonic Sensor, Temperature Sensor, Moisture Level Sensor)
Unit 11	Cloud and Data Analytics: Cloud storage models for IoT, Data Analytics for IoT, Amazon Web Services for IoT
Unit 12	Real-World IoT Constraints: Real-world design constraints and challenges, Case studies in major industries
Unit 13	Smart Applications: Applications: Home Automation, Industrial Automation, Smart Grid, Smart Agriculture, IoMT, Commercial Building Automation
Unit 14	IoT System Integration: Process of IoT System Integration, Integrating IoT systems with other technologies and applications

TEXTBOOKS:

- IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things Paperback – 20 June 2017.
- Arshdeep Bahga, Vijay Madisetti, “Internet of Things – A hands-on approach”, Universities Press, 2015.
- Rajkamal,”Internet of Things”, Tata McGraw Hill publication, 2nd Edition, 2022.

REFERENCE BOOKS:

- Olivier Hersent, David Boswarthick, Omar Elloumi , “The Internet of Things – Key applications and Protocols”, Wiley, 2012.
- Microcontroller and Embedded Systems”, Muhammad Ali Mazidi, Janice G. Mazidi, Rolin D. McKinlay, Pearson Education, Second edition, 2014.

DAYANANDA SAGAR UNIVERSITY

Course Name: PROBABILITY AND STATISTICS USING R

Credits: 4

Course Description:

This course comprehensively introduces the foundational concepts of probability, statistics, and data analysis, complemented by practical R programming applications. Students will learn essential statistical techniques such as measures of central tendency, dispersion, and correlation, enabling them to interpret and summarise data effectively. The course delves into probability distributions, sampling methods, and estimation theory, laying a strong foundation for hypothesis testing and inferential statistics. Furthermore, it introduces regression analysis and provides hands-on experience applying statistical tools to solve real-world problems. Through this curriculum, learners gain theoretical understanding and practical skills in data analysis.

Course Objectives:

- Outline the principles of probability and their application in statistics
- Discuss the descriptive and inferential statistical methods
- Describe the foundational concepts of R programming skills for data analysis
- Explore probability distributions and their real-world applications
- Interpret the statistical modelling techniques, including regression and hypothesis testing

Course Outcomes:

At the end of the course, the learner will be able to:

- Demonstrate proficiency in basic probability and statistical concepts
- Interpret data using measures of central tendency and dispersion
- Utilise R programming for statistical computations and visualisations
- Apply probability distributions and sampling techniques to real-world scenarios
- Conduct hypothesis tests and build regression models for predictive analysis

Unit 1	Basics of Probability Theory: Definitions, Probability Rules, Conditional Probability, Independent Events
Unit 2	Introduction to Statistics: Types and Uses: Descriptive vs. Inferential Statistics, Data Types (Nominal, Ordinal, Interval, Ratio)
Unit 3	Setting Up and Basics of R Programming: R Installation, IDEs (RStudio), Syntax Basics, Writing Scripts
Unit 4	Data Types and Data Structures in R: Vectors, Lists, Matrices, Data Frames, Factors
Unit 5	Measures of Central Tendency: Mean, Median, Mode, Applications in Data Summarization
Unit 6	Measures of Dispersion: Variance, Standard Deviation, Range, Interquartile Range
Unit 7	Random Variables and Probability Distributions: Discrete and Continuous Random Variables, Distribution Functions
Unit 8	Binomial and Poisson Distributions: Properties, Formulas, Real-life Applications of Binomial and Poisson Distributions
Unit 9	Normal Distribution and Standardization: Properties, Z-Scores, Applications
Unit 10	Sampling Distributions: Sampling Methods, Central Limit Theorem
Unit 11	Estimation Theory (Point and Interval Estimates): Confidence Intervals, Estimation Techniques
Unit 12	Hypothesis Testing Concepts and Applications: Null and Alternative Hypotheses, Types of Errors, P-Values
Unit 13	Correlation Analysis and Coefficient Interpretation: Pearson and Spearman Correlation, Correlation Coefficient
Unit 14	Simple Linear and Multiple Regression Analysis: Regression Equation, Interpretation of Coefficients, Model Evaluation

REFERENCE BOOKS:

- Seema Acharya, Data Analytics using R, McGrawHill Education (India), 2018, ISBN: 978-93-5260-524-8.
- Basic Statistics –Agarwal B.L, New age international, 6th edition, 2013.
- Medhi J – Statistical methods, New age international, Second edition, Reprint 2013.
- Walpole, Myers et al. –Probability and statistics for scientists and engineers., Pearson Education, Ninth edition, 2013.
- Applied Statistics and probability for Engineers, Runger and Montgomery, Wiley, 6th Edition.

Course Name: COMPUTER GRAPHICS AND MULTIMEDIA

Credits: 4

Course Description:

This course explores the fundamentals of computer graphics, multimedia systems, and their real-world applications. Learners will learn about video display devices, raster and random scan systems, and basic output primitives like DDA algorithms for points and lines. Topics include circle and ellipse generation, curve functions, and transformation techniques using matrices. Viewing pipelines, clipping operations, and colour models are introduced to enhance visualisation. The course also covers multimedia systems, databases, compression techniques (image, audio, and video), and animation design. Practical concepts like raster animation, motion specifications, and virtual reality applications are discussed to equip learners with essential graphics and multimedia skills.

Course Objectives:

- Gain knowledge about graphics hardware devices and software used
- Discuss the two-dimensional graphics and their transformations
- Describe the three-dimensional graphics and their transformations
- Appreciate illumination and colour models
- Explain the clipping techniques

Course Outcomes:

At the end of the course, the learner will be able to:

- Design and apply two-dimensional transformations using DDA, a geometric shape-generating algorithm
- Discuss three-dimensional transformations and clipping
- Apply the various techniques of compression in multimedia graphics
- Analyse file format standards and authoring systems
- Interpret color models in computer animations

Unit 1	Introduction to Computer Graphics: Survey of computer graphics, Visualization, GUI, Applications
Unit 2	Graphics Systems Overview: Video display devices, Raster and random scan systems, Graphics software, Monitors and workstations
Unit 3	Basic Output Primitives: Introduction, Points, and Lines, DDA Algorithm
Unit 4	Circle and Ellipse Generation: Circle-Generating Algorithms, Ellipse-Generating Algorithms
Unit 5	Curve Functions and Attributes: Parallel Curve Algorithms, Curve Functions, Attributes
Unit 6	Transformations and Matrices: Basic transformations, Matrix representations, Composite and other transformations
Unit 7	Viewing and Clipping: The viewing pipeline, Clipping operations (Point, Line, Polygon, Curve, Text, Exterior)
Unit 8	Multimedia Systems: Introduction, Applications, System Architecture, Evolving Technologies, Multimedia data interface standards
Unit 9	Multimedia Databases: Defining objects for multimedia systems, Multimedia databases
Unit 10	Compression Techniques: Types of Compression, Binary Image Compression, Color/Gray Scale/Still-Video Compression
Unit 11	Audio and Video Compression: Video image compression, Audio compression, Techniques, Case studies
Unit 12	Color Models and Light Properties: Properties of light, Standard primaries, Intuitive color concepts, Color models
Unit 13	Color Selection and Applications: Color selection, Applications in graphics and multimedia, case studies, ethical standards
Unit 14	Computer Animation: Design of animation sequences, Raster animation, Key-frame systems, Morphing, Motion specifications, Virtual Reality case study

TEXT BOOKS:

- Donald Hearn and Pauline Baker M, “Computer Graphics C Version”, Pearson Education, New Delhi, 2nd Edition, 2018.
- Foley, Vandam, Feiner and Huges, “Computer Graphics: Principles & Practice”, Pearson Education, Asia, 3rd Edition, 2018.

REFERENCE BOOKS:

- Prabat K Andleigh and Kiran Thakrar, “Multimedia Systems Design”, Prentice Hall of India, New Delhi 2015.
- Judith Jeffcoate, “Multimedia in practice technology and Applications”, Prentice Hall of India, New Delhi, 2009.

Course Name: LIBERAL STUDIES - I

Credits: 2

Course Description:

This course introduces fundamental economic principles, including demand-supply dynamics, firm theories, and market structures. Learners will explore macroeconomic concepts like GDP, GNP, national income, taxation systems, and aggregate demand-supply frameworks (IS/LM). Monetary systems, central bank functions, and commercial banking are discussed in detail. Key cost control and budgeting techniques such as break-even analysis, capital budgeting, and linear programming are covered. Investment analysis tools like NPV, ROI, IRR, and time value of money are explored for financial decision-making. The course concludes with an overview of the Indian economy, addressing post-reform growth, employment sectors, and contemporary policy debates.

Course Objectives:

- Discuss the economic principles, demand, and market structures
- Analyse macroeconomic concepts, aggregate demand, and supply
- Apply cost control, budgeting, and investment analysis techniques

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain the principles of economics, including demand, supply, and basic macroeconomic concepts
- Analyse price indices, public sector economics, monetary and fiscal policies, and capital markets
- Examine the Indian economy's challenges, policies, and debates in monetary, fiscal, social, and external sectors

Unit 1	Basic Principles of Economics: Basic Principles and Methodology of Economics, Demand/Supply – Theory of the Firm, Market Structure
Unit 2	Macroeconomic Concepts: Basic Macroeconomic Concepts (GDP, GNP, NI, Disposable Income), Identities for closed and open economies
Unit 3	Aggregate Demand and Supply: Aggregate Demand and Supply (IS/LM), Price Indices (WPI/CPI), Interest Rates, Direct and Indirect Taxes
Unit 4	Monetary and Financial Systems: Components of Monetary and Financial System, Central Bank – Monetary Aggregates, Functions of Commercial Banks
Unit 5	Cost Control and Budgeting: Cost & Cost Control Techniques, Types of Costs, Budgets, Break-even Analysis, Capital Budgeting, Linear Programming
Unit 6	Investment Analysis: NPV, ROI, IRR, Payback Period, Depreciation, Time Value of Money, Business Forecasting – Elementary Techniques
Unit 7	Indian Economy Overview: Post-independence plans, Post-reform growth, Employment (Informal, Organized, Unorganized, Public, Private sectors), Policy Debates

TEXTBOOKS:

- Mankiw Gregory N., Principles of Economics, Thompson Asia, 7th Edition, 2015.
- V. Mote, S. Paul, G. Gupta, Managerial Economics, Tata Mc Graw Hill, 2017.

REFERENCE BOOKS:

- Misra, S.K. and Puri, Indian Economy, Himalaya Publishing House,42nd Edition, 2024.
- Pareek Saroj, Textbook of Business Economics, Sunrise Publishers, 2009.

Semester: V

Course Name: INFORMATION AND NETWORK SECURITY

Credits: 3

Course Description:

This course explores the fundamental principles of computer security, covering threats, cryptographic techniques, and secure system design. Learners will learn about access control mechanisms, secure coding practices, and intrusion detection systems. Topics include classical and modern cryptography, public-key infrastructure, authentication protocols, malware analysis, and network security fundamentals. Learners will develop skills to identify vulnerabilities, design countermeasures, and implement effective security mechanisms to protect systems and networks. By the end of the course, learners will be equipped to address current challenges in cybersecurity and contribute to safeguarding digital systems through research and practical applications.

Course Objectives:

- Discuss the computer security principles and threats
- Explore cryptographic techniques and secure system design
- Apply secure coding practices and access control mechanisms
- Analyse network security, intrusion detection, and malware

Course Outcomes:

At the end of the course, the learner will be able to:

- Implement basic security algorithms and solutions for computing systems
- Analyse system vulnerabilities and security attacks and derive effective countermeasures
- Identify and resolve network security issues while evaluating security mechanisms
- Formulate research problems in the field of computer security.

Unit 1	Computer Security Overview: Overview of computer security, Security services, Security mechanisms, Security attacks
Unit 2	Security Policies and Access Control: Access control matrix, Security policies (Confidentiality, Integrity, Hybrid policies)
Unit 3	Cryptographic Techniques: Classical cryptography (Substitution, Permutation ciphers), Block ciphers, DES Modes of Operation, AES
Unit 4	Hash Functions and Authentication Protocols: Hash functions (SHA 512), Message Authentication Codes (HMAC), Authentication protocols
Unit 5	Public Key Cryptography Basics: Introduction to public key cryptography, Number theory, RSA Cryptosystem, Digital Signature Algorithm
Unit 6	Key Management and Cryptography: Key management (session/interchange keys, key exchange, PKI), Attacks on RSA, Elliptic Curves Cryptography
Unit 7	Secure System Design: Design principles, representing identity, access control mechanisms, information flow and confinement
Unit 8	Secure Coding Practices: Secure software development (OWASP/SANS vulnerabilities), Buffer overflows, Command injection, XSS
Unit 9	Network Security Fundamentals: Secret sharing schemes, Kerberos, PGP, SSL
Unit 10	Intrusion Detection and Malware: Intruders (HIDS, NIDS), Firewalls, Viruses

TEXTBOOKS:

- William Stallings, “Cryptography and Network Security: Principles and Practices”, 8th Edition, Pearson Education, 2022.
- Matt Bishop, “Computer Security art and science ”, 2nd Edition, Pearson Education, 2018.

REFERENCE BOOKS:

- Wade Trappe and Lawrence C. Washington, “Introduction to Cryptography with Coding Theory” 3rd Edition, Pearson Education, 2021.
- Jonathan Katz, and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 3rd Edition, 2021.
- Douglas R. Stinson, “Cryptography Theory and Practice”, Third Edition, Chapman & Hall/CRC, 2nd Edition, 2014.
- Wenbo Mao, “Modern Cryptography – Theory and Practice”, Pearson Education, 1st Edition, 2006.
- Network Security and Cryptography, Menezes Bernard, Cengage Learning, New Delhi, 2011.

Course Name: DATA MINING

Credits: 4

Course Description:

This course introduces the concepts, techniques, and applications of data mining. Learners will explore data preparation tasks, including cleaning, transformation, and integration, followed by analysis using classification, clustering, and association rule mining. Topics include decision trees, K-Means clustering, Naive Bayes classification, and advanced association techniques like FP growth. Learners will also study real-world case studies and emerging trends in data mining, such as big data and automated analytics. By the end of the course, learners will be able to analyse large datasets, extract meaningful patterns, and solve practical problems across various domains using data mining algorithms.

Course Objectives:

- Gain a foundational understanding of data mining processes and applications.
- Perform data preprocessing tasks like cleaning, integration, and transformation.
- Apply classification and clustering techniques using real-world datasets.
- Identify and analyse patterns in data using association rule mining.
- Explore advanced topics and challenges in data mining, such as scalability and ethical considerations.

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the fundamentals of data mining and its applications
- Apply data preparation techniques for practical data mining
- Utilise statistical and visualisation techniques to explore datasets
- Implement basic and advanced data mining algorithms
- Analyse trends and challenges in modern data mining

Unit 1	Overview of Data Mining: Definition, Importance, Data Mining vs. Other Analytical Techniques
Unit 2	Data Mining Process and Applications: CRISP-DM Process, Real-world Applications
Unit 3	Data Cleaning and Transformation: Handling Missing Values, Data Normalization, Feature Selection
Unit 4	Data Integration and Reduction: Combining Datasets, Dimensionality Reduction Techniques
Unit 5	Data Visualisation Techniques: Histogram, Boxplot, Scatter Plot, examples, case studies
Unit 6	Statistical Measures: Central Tendency, Variability, Correlation Analysis, case studies
Unit 7	Decision Trees: Building and Interpreting Decision Trees, Pruning
Unit 8	Naive Bayes Classifier: Bayesian Probability Concepts, Building a Naive Bayes Model
Unit 9	Introduction to Clustering; Concepts of Partitioning and Hierarchical methods, Types and significance, Partitioning vs. Hierarchical Methods, Use Cases
Unit 10	K-Means Clustering: Concepts of K-Means Clustering, Algorithm Steps, Choosing K, Limitations
Unit 11	Hierarchical Clustering: Concepts of hierarchical clustering, General process of hierarchical clustering, Agglomerative vs. Divisive Methods, Dendrograms
Unit 12	Market Basket Analysis: Apriori Algorithm, Support, Confidence, Lift Measures, examples and case studies
Unit 13	Advanced Association Techniques: Concepts of advanced association techniques, FP-Growth, ECLAT Algorithms, examples and case studies
Unit 14	Data Mining Trends and Challenges: Big Data and Data Mining, Ethical Issues, Trends in Automated Analytics

TEXTBOOKS:

- Han, J., Pei, J., & Kamber, M. Data Mining: Concepts and Techniques. Morgan Kaufmann, 4th Edition, 2022, ISBN: 978-0128112485.
- Aggarwal, C. C. Data Mining: The Textbook. Springer, 2015, ISBN: 978-3319141428

REFERENCES BOOKS:

- Witten, I. H., Frank, E., & Hall, M. A. Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann 4th Edition, 2016.

Course Name: MOBILE PROGRAMMING

Credits: 4

Course Description:

This course focuses on the fundamentals of mobile application development, specifically focusing on Android platforms. Learners will learn mobile project structure, UI design, and core components like layouts, activities, and intents. Topics include Android architecture, SQLite data persistence, and Google API integration for maps, messaging, and cloud services. Advanced features such as sensors, notifications, and performance optimisation are also covered. Through practical case studies, learners will conceptualise, develop, and deploy mobile applications using Android Studio. Learners will then learn to create user-friendly and functional mobile apps for real-world use.

Course Objectives:

- Discuss the basic concepts of mobile computing and Mobile platform
- Be familiar with the Android Studio
- Mobile app development using Android
- Learn the basics of Android Components
- Gain knowledge about different mobile platforms and application development through case studies.

Course Outcomes:

At the end of the course, the learner will be able to:

- Describe the new trends in mobile/wireless communications networks
- Demonstrate their ability to develop software with reasonable complexity on mobile platforms
- Develop UI-based Mobile Applications using Android Studio
- Apply the leading UI components, including layouts, widgets and views, to build a mobile application
- Demonstrate their skills in using Android studio

Unit 1	Mobile Communication Basics: Introduction, Characteristics, Advantages of mobile communication, Types of mobile applications
Unit 2	Mobile Project Structure: Mobile computing project structure, Mobile Device Operating Systems, Special Constraints & Requirements
Unit 3	SDKs and M-Commerce: Software Development Kits (iOS, Android, BlackBerry, Windows Phone), M-Commerce structure, Pros & Cons, Mobile Payment Systems
Unit 4	Android Overview: Introduction to Android, Trends, Platforms, Development Setup (Android Studio, Eclipse, Android SDK, tools), Emulator setup
Unit 5	Android Architecture: Android Runtime (ART), Android Architecture
Unit 6	App Development Basics: Android App Use Case, Use Case Diagram (UML), User Interface Implementation, Values, Android Manifest File
Unit 7	Activities and Intents: Activities, Activity Lifecycle, Intents (Types of Intents), UI Design
Unit 8	Android Layouts and Views: Android Components - layouts, fragments, basic views, list views, picker views, adapter views, menus, action bars
Unit 9	Data Persistence: Managing data using SQLite, User content provider, Persisting data into database
Unit 10	Google APIs for Android: Using Google APIs (Maps, Cloud Messaging, Authentication, Storage, Hosting, Google Play services)
Unit 11	Mobile Application Case Study: Developing a mobile application using Android Studio (conceptualization, development, testing)
Unit 12	UI/UX Considerations: Focus on User Interface and User Experience design in mobile apps
Unit 13	Advanced Features: Integrating advanced features such as device sensors, notifications, and user settings
Unit 14	Deployment and Distribution: Deploying apps on Google Play Store, App testing strategies, Performance optimization

TEXTBOOKS:

- Debashis De, Mobile Cloud Computing Architectures, Algorithms and Applications, CRC Press, 1st Edition, 2016.
- Adem Karahoca, Advances and Applications in Mobile Computing, InTechOpen, 2012, ISBN 978-953-51-0432-2.
- Marko Gargenta, Learning Android, Printed in the United States of America, O'Reilly Media Inc, 2011.

- Reto Meier, Professional Android™ 2 Application Development, Wrox, 1st Edition, 2010.

REFERENCE BOOKS:

- Jochen H. Schller, “Mobile Communications”, Second Edition, Pearson Education, New Delhi, 2008.
- Dharma Prakash Agarval, Qing and An Zeng, "Introduction to Wireless and Mobile systems", Thomson Asia Pvt Ltd, 2005.

DAYANANDA SAGAR UNIVERSITY

Course Name: SOFTWARE PROJECT MANAGEMENT

Credits: 4

Course Description:

This course covers the principles, methodologies, and practices of software project management. Learners will learn to plan, estimate, and schedule projects effectively while addressing resource allocation, risk mitigation, and communication strategies. Topics include project lifecycle management, Gantt charts, critical path methods, and quality assurance processes like CMMI and Six Sigma. The course also focuses on conflict resolution, change management, and project closure techniques to ensure successful project delivery. By integrating real-world case studies, learners will develop the skills to manage software projects efficiently, overcome challenges, and deliver high-quality solutions.

Course Objectives:

- Discuss the principles of software project management and the role of a project manager.
- Learn to plan and estimate project scope, resources, and schedules effectively.
- Apply risk management techniques to identify, assess, and mitigate risks
- Ensure project quality through control measures and process improvement models
- Monitor and control project progress, handle changes, and close projects successfully

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the fundamentals of project management and its life cycle
- Develop project plans and estimation strategies
- Identify potential risks, assess their impact, and develop appropriate mitigation strategies
- Implement quality assurance and process improvement models
- Develop communication plans, handle change requests, and ensure smooth project closure

Unit 1	Basics of Project Management: Definition, Importance, Project Life Cycle, Project vs. Product
Unit 2	Software Project Management Overview: Role of a Project Manager, Skills Required, Overview of Process Models
Unit 3	Project Planning and Estimation: Project Scope, Objectives, Work Breakdown Structure (WBS), Resource Allocation
Unit 4	Scheduling and Cost Estimation: Gantt Charts, PERT, Critical Path Method (CPM), Budget Estimation
Unit 5	Identifying and Assessing Risks: Types of Risks, Risk Identification Techniques, Case studies
Unit 6	Risk Mitigation Strategies: Risk Response Planning, Monitoring and Controlling Risks
Unit 7	Quality Assurance and Control: Quality Metrics, Standards, Reviews, Testing Techniques
Unit 8	Process Improvement Models: Capability Maturity Model Integration (CMMI), Six Sigma
Unit 9	Human Resource Management in Projects: Team Building, Conflict Resolution, Resource Allocation Techniques
Unit 10	Communication Management: Communication Plan, Stakeholder Communication, Process for Stakeholder communication
Unit 11	Project Monitoring Techniques: Milestone Tracking, Progress Reports, Performance Metrics
Unit 12	Project Change Management: Change Requests, Impact Analysis, Change Control Process
Unit 13	Project Closure Process: Closure Report, Client Acceptance, Post-mortem Analysis
Unit 14	Project Success and Failure Analysis: Critical Success Factors, Lessons Learned, Avoiding Common Pitfalls

TEXTBOOKS:

- Larson, E., & Gray, C. *Project Management: The Managerial Process*. McGraw-Hill Education, 2021, 8th Edition, ISBN: 978-1260238860.
- Schwalbe, K. (2020). *Information Technology Project Management*, Cengage Learning, 9th Edition, ISBN: 978-0357133490.

REFERENCE BOOKS:

- Project Management Institute (PMI). *A Guide to the Project Management Body of Knowledge (PMBOK Guide)* (7th ed.). Project Management Institute, 2021.
- Kerzner, H., *Project Management: A Systems Approach to Planning, Scheduling, and Controlling* (12th ed.). Wiley, 2017.

DAYANANDA SAGAR UNIVERSITY

Course Name: BLOCKCHAIN TECHNOLOGY

Credits: 4

Course Description:

This course explores blockchain technology, cryptographic foundations, and distributed ledger concepts. Learners will study blockchain components such as blocks, consensus mechanisms (Proof-of-Work and Proof-of-Stake), and peer-to-peer networks. The course introduces smart contracts and their deployment using Solidity and blockchain platforms like Bitcoin and Ethereum. Topics also include blockchain applications in finance, healthcare, and supply chain industries, along with scalability, security, and regulation challenges. By the end of the course, learners will be proficient in developing essential blockchain solutions, understanding decentralised systems, and evaluating the potential and limitations of blockchain technologies.

Course Objectives:

- Discuss the basics of blockchain technology, its components, and applications
- Learn cryptographic foundations, including hashing and public-key cryptography
- Explore various consensus mechanisms such as Proof-of-Work and Proof-of-Stake
- Develop and deploy smart contracts using blockchain platforms
- Interpret the current challenges and future trends in blockchain systems

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the fundamentals of blockchain technology and its applications
- Analyse the components and cryptographic foundations of blockchain systems
- Compare distributed and centralised systems in the context of blockchain
- Develop and deploy basic smart contracts
- Evaluate blockchain use cases and address challenges in the implementation

Unit 1	Overview of Blockchain Technology: Definition, History, Key Characteristics, Applications
Unit 2	Components of Blockchain: Blocks, Transactions, Chains, Consensus Mechanisms
Unit 3	Cryptographic Hash Functions: Hashing, SHA-256, Properties, Use Cases
Unit 4	Public and Private Key Cryptography: Asymmetric Encryption, Digital Signatures
Unit 5	Distributed vs. Centralized Systems: Benefits, Challenges, Blockchain as Distributed Ledger
Unit 6	Peer-to-Peer Networks: Network Architecture, Node Communication, Evolution, Types, Case studies
Unit 7	Proof-of-Work (PoW): Mining Process, Security Implications, Limitations
Unit 8	Proof-of-Stake (PoS) and Variants: Delegated PoS, Practical Byzantine Fault Tolerance (PBFT), Case studies
Unit 9	Introduction to Smart Contracts: Definition, History, Benefits, Types, Process to be followed, Pros and Cons
Unit 10	Writing Smart Contracts: Basics of Solidity Programming, Tools (e.g., Remix, Ganache)
Unit 11	Bitcoin: Protocol, Mining Process, Transaction Flow, Real-time examples
Unit 12	Ethereum: Differences from Bitcoin, Smart Contracts, dApps
Unit 13	Blockchain Use Cases: Supply Chain, Healthcare, Financial Services, Decentralized Finance (DeFi)
Unit 14	Challenges and Future Directions: Scalability, Security, Regulatory Issues, Emerging Trends

TEXTBOOKS:

- Mougayar, W. The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology. Wiley, 2021, ISBN: 978-1119365594.
- Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S., Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction. Princeton University Press, 2016, ISBN: 978-0691171692.

REFERENCES BOOKS:

- Wood, G. "Ethereum: A Secure Decentralised Generalised Transaction Ledger." Ethereum Yellow Paper, 2014.

DAYANANDA SAGAR UNIVERSITY

Course Name: DATABASE ADMINISTRATION

Credits: 4

Course Description:

This course provides a comprehensive understanding of database management and administration using Oracle. Learners will learn database installation, configuration, and maintenance while managing physical and logical structures. Topics include SQL operations, constraint management, user access, and security mechanisms. The course also addresses performance monitoring, backup, and recovery strategies for ensuring data reliability and integrity. Learners will explore table and index management, network design considerations, and advanced database operations. By the end of the course, learners will possess the skills to design, maintain, and optimise database systems effectively while addressing modern database challenges.

Course Objectives:

- Discuss the problems and appropriate solutions involved in the role of database management and administration
- Learn about the Installation, creation, configuration and maintaining an operational database
- Managing physical and logical database structures, including users, objects and data
- Monitoring performance and identifying performance improvements
- Planning and implementing backup/recovery strategies 6. Managing database security

Course Outcomes:

At the end of the course, the learner will be able to:

- Design, model and install any database management systems by using Oracle database as a sample
- Plan, design, construct, control and manage database instances, database network environment, storage structures, user security, database backup and recovery, and database maintenance.
- Define and devise transaction management, concurrency control, and crash recovery components.

- Examine and perform database administration roles and operations using the Oracle database system as a sample.
- Compare and contrast by examining the database systems and new trends in data storage, data retrieval and maintenance techniques.

Unit 1	DBMS Architecture and Roles: DBMS architecture, Data independence, DBA roles and responsibilities
Unit 2	SQL*Plus Fundamentals: SQL Plus overview, Producing readable outputs, Accepting values at runtime
Unit 3	Data Manipulation Basics: DML Statements, Truncating tables, Transaction control language
Unit 4	Constraint Management: Creating, dropping, enabling, disabling constraints, Deferring constraint checks
Unit 5	Managing Views: Creating, modifying, using views, Data manipulation through views
Unit 6	User Access and Security: Creating/modifying user accounts, Creating/using roles, Privilege management, User groups and profiles
Unit 7	Oracle Storage and Processes: Logical and physical storage structures, Memory structures, Background processes
Unit 8	Managing Oracle Instances: Connecting to instances, Processing SQL commands, Managing sessions, Starting/shutting down instances
Unit 9	Table and Index Management: Creating, altering, analyzing tables, Managing indexes, Constraints management
Unit 10	User and Privilege Management: Managing profiles, Users, Privileges, Roles
Unit 11	Network Administration Basics: Network design considerations, Responsibilities for DBA, Network configuration
Unit 12	Oracle Net Features: Overview of Oracle Net features, Net Stack Architecture
Unit 13	Backup and Recovery Basics: Database backup, Restoration and recovery, Types of failure in Oracle
Unit 14	Backup and Recovery Strategies: Defining and testing backup and recovery plans

TEXTBOOKS:

- C.J. Date, Database Systems, Addison Wesley, 2006.
- Chip Dawes, Biju Thomas, Introduction to Oracle 9i SQL, BPB, 2002.
- Bob Bryla, Biju Thomas, Oracle 9i DBA Fundamental I, BPB, 2002.
- Doug Stums, Matthew Weshan, Oracle 9i DBA Fundamental I, BPB, 2002.

REFERENCE BOOKS:

- Joseph C. Johnson, Oracle 9i Performance Tuning., BPB, 2002.

DAYANANDA SAGAR UNIVERSITY

Course Name: BIG DATA ANALYTICS

Credits: 4

Course Description:

This course introduces learners to big data concepts, their challenges, and applications in various industries. Learners will gain hands-on experience with programming in Scala, Apache Spark architecture, and NoSQL databases. The course covers data distribution, real-time processing, and advanced analytics techniques for handling massive datasets. Learners will also explore machine learning algorithms, Spark SQL, and data lifecycle management. Practical examples and case studies enable learners to understand big data's role in solving real-world problems like healthcare, e-commerce, and finance. By the end, learners will be proficient in applying big data technologies and analytics for decision-making.

Course Objectives:

- Discuss the big data concepts, characteristics, and challenges
- Explore Scala basics and Spark architecture
- Apply NoSQL and data distribution models
- Analyse big data using Spark and analytics tools
- Implement machine learning techniques in big data

Course Outcomes:

At the end of the course, the learner will be able to:

- Identify Big Data and its Business Implications
- Discuss the programming in Scala
- Learn Apache Spark and Bigdata Analytics
- Explain the basics of NoSQL
- Demonstrate the concepts of Data Analytics with Scala

Unit 1	Introduction to Big Data: Evolution, Definition, Challenges with Big Data, Characteristics of Data, Why Big Data
Unit 2	Types and Industry Examples: Classification of Digital Data (structured, unstructured, semi-structured), Big Data in Healthcare, Medicine, Advertising
Unit 3	Scala Overview: Overview, Benefits of Scala, Why Scala, Prerequisites, Installation of Scala
Unit 4	Scala Basics – I: Genesis and Characteristics of Spark, Spark Evolution, Spark Ecosystem
Unit 5	Scala Basics – II: Scala Building Blocks, Basics of the Language, Scala Classes and Methods (OOP, Case Classes, Method Definitions)
Unit 6	Spark Architecture and RDD: Apache Spark Architecture, RDD in Spark, Features of Spark, Simple Use cases
Unit 7	Advanced Spark Concepts: Spark Catalyst Optimizer, Spark SQL, Spark Application, SparkSession
Unit 8	Introduction to NoSQL: Why NoSQL, Aggregate Data Models (Key-Value, Document Models), Graph Databases, Schema-less Databases
Unit 9	Data Distribution and Processing: Distribution Models (Sharding, Versioning), Map-Reduce, Partitioning and Combining
Unit 10	Big Data Analytics Overview: Classification of Analytics, Challenges, Importance of Big Data Analytics, Applications, Real-time examples
Unit 11	Technologies for Analytics: Technologies for Big Data challenges, Apache Spark and Scala for Analytics
Unit 12	Big Data Analytics Lifecycle: Introduction to Analytics Lifecycle, Types of Analytics, Big Data Analytics Problems
Unit 13	Analytics Using Machine Learning: Big Data Analytics using Machine Learning Techniques
Unit 14	Big Data with Big R: Introduction to Big Data with Big R, Analytical capabilities with R in Big Data

TEXTBOOKS:

- Martin Odersky, Lex Spoon, Bill Venners, Programming in Scala, 5th Indian Edition, 2022.
- Dean Wampler and Alex Payne, Programming Scala, Oreilly Publication, 3rd edition, 2021.
- Jules S. Damji, Brooke Wenig, Tathagata Das, and Denny Lee, Learning Spark-Lightning-Fast Data Analytics, Oreilly Publication, 2020.
- Hoboken, NJ, Dan Sullivan, NoSQL for Mere Mortals, Addison Wesley, 2015.
- Pramod J. Sadalage, Martin Fowler, NoSQL Distilled-A Brief Guide to the Emerging World of Polyglot Persistence, Addison Wesley, 2012.

REFERENCE BOOKS:

- Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
- Jay Liebowitz, "Big Data and Business Analytics" Auerbach Publications, CRC press, 2013.
- Tom Plunkett, Mark Hornick, "Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop", McGraw-Hill/Osborne Media, Oracle press, 2013.
- Anand Rajaraman and Jef rey David Ulman, "Mining of Massive Datasets", Cambridge University Press, 2012.
- Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons, 2012.
- Glen J. Myat, "Making Sense of Data", John Wiley & Sons, 2007
- Pete Warden, "Big Data Glossary", O'Reily, 2011.
- Michael Mineli, Michele Chambers, Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley Publications, 2013.
- ArvindSathi, "BigDataAnalytics: Disruptive Technologies for Changing the Game", MC Press, 2012
- Paul Zikopoulos ,Dirk DeRoos , Krishnan Parasuraman , Thomas Deutsch , James Giles , David Corigan , "Harness the Power of Big Data The IBM Big Data Platform ", Tata McGraw Hill Publications, 2012.

DAYANANDA SAGAR UNIVERSITY

Course Name: OPENSOURCE DEVELOPMENT USING PHP AND MYSQL

Credits: 4

Course Description:

This course introduces learners to PHP and MySQL for developing dynamic, database-driven web applications. Topics include open-source principles, PHP syntax, control structures, form handling, and CRUD operations with MySQL. Learners will explore advanced features such as file handling, object-oriented programming in PHP, and web security essentials like SQL injection prevention. The course culminates in developing a functional web application incorporating user authentication, secure coding practices, and deployment techniques. By the end, learners will acquire the skills to design, develop, and maintain safe and efficient open-source web applications.

Course Objectives:

- Set up a PHP and MySQL development environment for open-source web applications
- Create and manipulate dynamic web pages using PHP and user input handling
- Build and interact with MySQL databases using PHP for CRUD operations
- Apply security measures to protect web applications from common threats
- Develop, deploy, and maintain a complete PHP and MySQL-based web application

Course Outcomes:

At the end of the course, the learner will be able to:

- Learn about open-source definitions, licenses, and their significance
- Develop dynamic web applications using PHP and MySQL
- Implement secure and efficient web application techniques
- Apply object-oriented programming concepts in PHP
- Plan, deploy, and maintain web applications

Unit 1	Overview of Open Source: Definition, Principles, Benefits, Open Source Licenses
Unit 2	Introduction to PHP and MySQL: Features, Use Cases, PHP vs. Other Languages, MySQL Overview
Unit 3	Setting Up PHP Environment: Installation (XAMPP, WAMP), Basic Syntax, Variables
Unit 4	Control Structures and Functions: Conditional Statements, Loops, Functions, Scope
Unit 5	Form Handling: GET and POST Methods, Form Validation, Data Sanitization
Unit 6	Sessions and Cookies: Managing Sessions, Setting and Retrieving Cookies
Unit 7	Database Connections: Connecting to MySQL, Basic Queries, CRUD Operations
Unit 8	Advanced Database Operations: Joins, Transactions, Prepared Statements
Unit 9	File Handling: Reading and Writing Files, File Uploads
Unit 10	Object-Oriented PHP: Classes, Objects, Inheritance, Encapsulation
Unit 11	Web Security Essentials: SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF)
Unit 12	Data Validation and Sanitization: Input Validation Techniques, Sanitization Functions
Unit 13	Developing a Basic Web Application: Planning, Building, Connecting PHP with MySQL, User Authentication
Unit 14	Deployment and Maintenance: Hosting PHP Applications, Maintenance Practices, Real-time examples

TEXTBOOKS:

- Ullman, L. *PHP and MySQL for Dynamic Web Sites* (5th ed.). Peachpit Press. 2021, ISBN: 978-0134301846.
- Holzner, S. *PHP: The Complete Reference* (2nd ed.). McGraw-Hill Education. 2019, ISBN: 978-1260457681.

REFERENCE BOOKS:

- Welling, L., & Thomson, L. *PHP and MySQL Web Development* (6th ed.). Addison-Wesley Professional, 2021.

- Fowler, M. "Best Practices for Secure PHP Development." *ACM Transactions on Web Programming*, 2020.

Course Name: FULL STACK DEVELOPMENT

Credits: 4

Course Description:

This course provides in-depth knowledge of full-stack development, combining front-end and back-end technologies. With MongoDB, learners will learn server-side programming using Node.js, Express, and database operations. Topics include handling HTTP requests, user authentication, and data storage techniques. For front-end development, learners will explore React.js, state management, and Redux concepts for building dynamic, user-centric applications. Through hands-on practice, learners will develop two-tier and three-tier applications, integrate APIs, and deploy real-world web solutions. By the end, learners will gain the expertise to design and implement scalable, interactive, and responsive full-stack web applications.

Course Objectives:

- To become knowledgeable about the most recent web development technologies
- Creating two-tier and three-tier architectural web applications
- Interpret real-time web applications
- Constructing suitable client and server-side applications
- Usage of core concepts of both front-end and back-end programming

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain the role and importance of Node.JS
- Outline server-side programming techniques
- Explain NoSQL and MongoDB programming concepts
- Interpret and learn REACT.JS
- Develop client-side web applications using React.JS

Unit 1	Node.JS Basics: Installation, Example NodeJS application, Fundamentals of NodeJS
Unit 2	NodeJS vs Browser: Differences between NodeJS and the browser, Techniques, Standard working procedures
Unit 3	Web Servers and NodeJS: Introduction to web servers, Running JavaScript in the desktop with NodeJS
Unit 4	Express Framework and NPM: Serving files with the http module, Introduction to Express, NPM usage, Templating engines, Static files, async/await
Unit 5	Fetching Data and JSON Handling: Process, Significance, Fetching JSON data from Express
Unit 6	NoSQL and MongoDB Basics: Introduction to NoSQL databases, MongoDB overview, Basic querying with MongoDB shell
Unit 7	NodeJS and Database Operations: Request body parsing in Express, NodeJS MongoDB connection, Adding/retrieving data with NodeJS
Unit 8	SQL and User Authentication: Handling SQL databases, Cookies in NodeJS, User Authentication with NodeJS
Unit 9	Introduction to React: Introduction to React, React Router, Single Page Applications
Unit 10	React Forms and Redux Basics: React Forms, Flow Architecture, Introduction to Redux, Client-server communication
Unit 11	Advanced Redux Concepts: More Redux features, Client-server communication, case studies
Unit 12	ReactDOM and JSX: ReactDOM, JSX, Components, Properties
Unit 13	State Management and Lifecycle: Fetch API, State and Lifecycle management in React, Events, Local storage
Unit 14	Composition and Advanced Concepts: Lifting State Up, Composition, Inheritance

TEXTBOOKS:

- Alex Banks, Eve Porcello , “Learning React”, O’ReillyMedia , Inc, 7th edition, 2020.
- Marc Wandschneider, “Learning Node”, Addison-Wesley Professional, 2nd edition, 2016.
- Azat Mardan, Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB. 2018, Apress.

REFERENCE BOOKS:

- Matee Frisbie, “ Java Script for Web Developers “, Wiley Publishing,Inc , 4th Edition, 2019, ISBN:9781-119-36656-0.

- Tomasz Dyl Kamil Przeorski , Maciej Czarnecki, Mastering Full Stack React Web Development Paperback – April 28, 2017, Packt Publishing Limited.

Course Name: LIBERAL STUDIES - II

Credits: 1

Course Description:

This course introduces learners to design thinking as a structured approach to innovation and problem-solving. Learners will use empathy-based methods to define user-centric problems, ideate solutions, and develop prototypes. Topics include brainstorming techniques, systems thinking, and iterative prototyping processes, emphasising feedback loops and refinement. By exploring real-world examples, learners will gain practical experience delivering impactful solutions focusing on user experience and ergonomic challenges. Through presentations and collaborative exercises, the course encourages creativity, critical thinking, and teamwork for addressing complex problems with innovative design solutions.

Course Objectives:

- Discuss the fundamentals of design thinking and its application to innovation.
- Apply ideation, prototyping, and feedback techniques to develop and refine user-centric solutions.

Course Outcomes:

- Demonstrate a clear understanding of the design thinking process and problem definition.
- Employ brainstorming, prototyping, and presentation techniques to deliver impactful, user-focused solutions.

Unit 1	Basics of Design Thinking: Innovation in Business, Creative Thinking & Problem Solving, Customer-Centricity in Design Thinking, Real World Examples (Airbnb, Apple, IDEO, Netflix), The Four Stages of Design Thinking: Empathize, Define, Ideate, Prototype, Implement.
Unit 2	Learning to Empathise and Define the Problem: Importance of Empathy in Innovation, Developing Empathy Using Design Tools, Observing and Assimilating Information, Individual Differences and Wicked Problems, Identifying Wicked Problems and Their Impact.

Unit 3	Ideate, Prototype, and Implement: Templates for Ideation (Brainstorming, Systems Thinking), Reaching Consensus on Wicked Problems, Mapping Customer Experience, Prototyping Methods, Rapid Prototyping, Implementing Design Ideas.
Unit 4	Feedback, Re-Design & Re-Create: Feedback Loops, User Experience Focus, Ergonomic Challenges, User-Focused Design, Concept Testing, Final Presentations of Innovative Design Solutions, Refining Solutions through Feedback and Iteration.

TEXTBOOKS:

- E Balaguruswamy, Developing Thinking Skills (The way to Success), Khanna Book Publishing Company, 2023.
- Tim Brown, “Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation”, Harvard Business Review, 2008.
- R T Krishnan & V Dabholkar, 8 steps to Innovation, Collins Publishing, 2013.

Semester: VI

Course Name: MACHINE LEARNING

Credits: 4

Course Description:

This course provides a practical and theoretical foundation in machine learning concepts, techniques, and tools. Learners will explore supervised and unsupervised learning algorithms, including k-nearest Neighbours, linear models, decision trees, and clustering techniques like K-means. Topics include data collection, preprocessing, visualisation, and model training using Python libraries. Learners will apply machine learning to problems such as classification, segmentation, and semi-supervised learning. Through case studies and hands-on practice, learners will gain the skills to develop robust machine-learning models, enabling them to address challenges in data-driven applications across industries.

Course Objectives:

- Discuss the fundamentals, challenges, and applications of machine learning
- Explore Python tools, libraries, and techniques for implementing machine learning
- Learn data collection, visualisation, and preparation processes for ML algorithms
- Study supervised and unsupervised learning techniques, including classification, regression, and clustering
- Apply advanced ML models and clustering techniques to real-world scenarios and case studies

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain the core concepts, types, and challenges of machine learning systems
- Utilise Python tools and libraries to collect, prepare, and visualise data ethically
- Implement supervised learning models like k-Nearest Neighbors, linear models, decision trees, and Naive Bayes
- Apply unsupervised learning techniques, including K-Means and advanced clustering algorithms

- Demonstrate practical applications of ML techniques through case studies, including image segmentation and semi-supervised learning.

Unit 1	Introduction to Machine Learning: What is Machine Learning?, Why Use Machine Learning?, Types of Machine Learning Systems
Unit 2	Challenges and Applications: Main challenges of Machine Learning, Applications of Machine Learning
Unit 3	Python and Tools for ML: Why Python for ML?, scikit-learn, Essential Libraries and Tools
Unit 4	Working with Real Data: Understanding real data, Big picture perspective, Types of Real data, Steps and architecture
Unit 5	Data Collection and Visualization: Process of data collection, Rules to be followed while collecting data, Ethical standards, Getting data, Discovering and visualizing data
Unit 6	Data Preparation for ML: Steps for Preparing data for ML algorithms, Ethical standards, examples and case studies
Unit 7	Model Selection and Training: Steps for Selecting and training a model, Rules to be followed, case studies
Unit 8	Introduction to Supervised Learning: Classification and Regression, Sample datasets, examples and case studies
Unit 9	k-Nearest Neighbors and Linear Models: k-Nearest Neighbors algorithm, Linear models, examples and case studies
Unit 10	Decision Trees and Naive Bayes: Decision trees, Naive Bayes classifiers, examples and case studies
Unit 11	Introduction to Unsupervised Learning: Types of unstructured learning, Clustering and its applications, examples and demonstrations
Unit 12	K-Means and Its Applications: K-Means clustering, Limits of K-Means, Image segmentation
Unit 13	Advanced Clustering Techniques: DBSCAN, Other clustering algorithms, examples and case studies
Unit 14	Clustering Applications: Using clustering for preprocessing and semi-supervised learning

TEXTBOOKS:

- Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Fourth Edition, 2020.
- Stephen Marsland, “Machine Learning: An Algorithmic Perspective, 2nd Edition, CRC Press, 2014.

REFERENCE BOOKS:

- Christopher M. Bishop, “Pattern Recognition and Machine Learning”, Springer, 2006.
- Tom Mitchell, “Machine Learning”, McGraw Hill, 3rd Edition, 1997.
- Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, “Foundations of Machine Learning”, Second Edition, MIT Press, 2018.
- Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2016.
- Sebastain Raschka, Vahid Mirjalili , “Python Machine Learning”, Packt publishing, 3rd Edition, 2019.

Course Name: GAME PROGRAMMING

Credits: 4

Course Description:

This course equips learners with the skills to design and develop interactive games using Unity. Topics include 2D game design fundamentals, asset management, sprite creation, and animation principles. Learners will learn coding for player controls, physics-based interactions, level design, and UI integration. Advanced topics cover prototyping, gameplay mechanics, and deployment strategies. The course also explores sound effects, HUD development, and optimisation techniques to enhance game performance. By building a mini-project, learners will gain hands-on experience in creating engaging games, preparing them for careers in game design and development.

Course Objectives:

- Discuss the unity interface for designing characters
- Develop game objects 2D Sprites efficiently
- Usage of layers, cameras, and colliders for animation
- Implement physics for 2D bodies
- Develop applications using the UI design. Build and deploy mini-project

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the concepts of Game design and development
- Design the processes and use mechanics for game development
- Explain the Core architectures of Game Programming
- Discuss Game programming platforms, frameworks and engines
- Outline and Define to Create interactive Games

Unit 1	Introduction to Unity 2D:Downloading and installing Unity, Project Wizard, Components, Game Objects
Unit 2	Project Setup and Interface:Creating a scene, Setting up a new project, Folder organization, Unity Interface, Android SDK
Unit 3	Unity Components and Tools:Player settings, Importing assets, File formats, Sounds, Performance, Transform, Camera, Mesh and Geometry
Unit 4	Raster and Vector Design:Creating raster and vector design, Vector illustration, Modular design, File formats
Unit 5	Working with Assets:Importing assets, Packages, GameObjects, Components, 2D Behaviors, Building sprites (character, environment, enemy, props)
Unit 6	Sprite Editing and Rendering:Sprite editor, External files, Sprite render, Props design
Unit 7	Level Design Basics:Level design fundamentals, Scene manipulation, Layered sorting, Prefabs
Unit 8	Coding and Animation:Player controller, Camera setup, Colliders, Animation principles, Creating animation states, State machines
Unit 9	Game Hierarchy and Assets:Game hierarchy, Asset management, Strtucture of assest management, real-time exmples
Unit 10	2D Physics and Settings:2D vs 3D settings, Rigid bodies, Colliders, Trigger points, Checkpoints, Player stats
Unit 11	Gameplay Mechanics:Scoring setup, Challenging gameplay elements (enemy controller, collisions, animations, damage, backgrounds)
Unit 12	Prototyping and Pathfinding:Prototyping, Pathfinding, Process of both, Architecture of protyping and path finding, case stduies
Unit 13	UI, HUD, and Audio Systems:UI Design, GUI, HUD elements, Particle system, Audio system
Unit 14	Optimization and Deployment:Organization, Optimization, Building, and Deploying games, UGUI

TEXTBOOKS:

- James A. Henley, Matthew Johnson, Learning 2D Game Development with Unity®: A Hands-On Guide to Game Creation - Addison-Wesley Professional, 2014.
- Venita Pereira, Learning Unity 2D Game Development, Packt Publishing Limited, 2014.
- Alan Thorn, Learn Unity for 2D Game Development, Apress Release Date: October 2013.
- Dr. Claudio Scolastici , Unity 2D Game Development Cookbook, Packt Publishing Limited, 2015.
- Patrick Felicia, A Beginner's Guide to 2D Platform Games with Unity: Create a Simple 2D Platform Game and Learn to Code in C# in the Process, Kindle Edition - Amazon Asia-Pacific Holdings Private Limited.
- Jeremy Gibson , Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#, Paperback – 2016.

- Shawn Nelson, Photoshop for Games: Creating Art for Console, Mobile, and Social Games, New Riders - 1 edition, 2014.

REFERENCE BOOKS:

- Fundamentals of Game Design - Ernest Adams - Pearson Education India – 3rd edition, 2015.
- Ashley Godbold , Simon Jackson, Mastering Unity 2D Game Development, Paperback – Import, 14 Oct 2016.
- Dave Calabrese, Unity-2d-game-development, Packt Publishing Limited, 18 March 2014.
- Chris Dickinson, Unity 5 Game Optimization, Packt Publishing Limited, 6 November 2015.
- The Animator's Eye: Adding Life to Animation with Timing, Layout, Design, Color and Sound, Focal Press; 1st Edition, 19 October 2012,
- Walt Disney Animation Studios The Archive Series #4: Layout & Background (Walt Disney Animation Archives) - Walt Disney Animation Research Libr (Author) - Disney Editions (25 October 2011)

Course Name: ETHICAL HACKING

Credits: 4

Course Description:

This course provides an in-depth understanding of ethical hacking techniques and tools to identify and counteract security threats. Learners will study hacking phases, footprinting, scanning, and enumeration methods to detect vulnerabilities. The course covers password cracking, privilege escalation, and penetration testing processes, focusing on ethical standards and real-world applications. Learners will gain hands-on experience using tools for network defence and attack analysis. By the end of the course, learners will develop the expertise to analyse security breaches, conduct penetration tests, and implement preventive measures to safeguard systems against evolving cyber threats.

Course Objectives:

- Analyse the fundamental concepts of information security
- Devise the various scanning tools and apply these methods in real-time applications
- Understand the enumeration tools and techniques
- Get the idea of system hacking with various tools and real-time applications
- Describe and illustrate the concept of Penetration testing tools and techniques

Course Outcomes:

At the end of the course, the learner will be able to:

- Articulate the fundamental concepts of information security and the complexity of the different attacks
- Defend hacking attacks on various scanning tools
- Describe the hacking attacks of various enumeration tools
- Analyse various hacking attack tools
- Apply appropriate technologies and approaches for penetrating tools and techniques

Unit 1	Basics of Hacking and Security:Introduction to hacking, Importance of security, Elements of security
Unit 2	Phases of an Attack:Phases of an attack, Types of hacker attacks, Hacktivism
Unit 3	Footprinting Techniques:Introduction to footprinting, Information gathering methodology, Footprinting tools
Unit 4	DNS and WHOIS Tools:WHOIS tools, DNS information tools, real-time samples, Working of WHOIS and DNS, Structure of WHOIS and DNS information extraction methods
Unit 5	Scanning Fundamentals:Introduction to scanning, Objectives, Scanning methodology
Unit 6	Scanning Tools:Tools for scanning, case studies
Unit 7	Basics of Enumeration:Introduction to enumeration, Objectives, Types of enumeration, Significance, case studies
Unit 8	Enumeration Techniques:Enumeration techniques and tools, Enumeration procedure
Unit 9	Password Cracking:Introduction, Cracking passwords, Password guessing, Password cracking websites, Tools
Unit 10	Privilege Escalation:Password cracking countermeasures, Escalating privileges
Unit 11	Introduction to Penetration Testing:Security assessments, Types of penetration testing, Evolution, Steps in penetration testing
Unit 12	Penetration Testing Phases:Phases of penetration testing with examples, Pros and Cons of penetration testing, ethical standards of penetration testing
Unit 13	Penetration Testing Tools:Tools used for penetration testing, categories, case studies
Unit 14	Practical Penetration Testing:Practical application and real-world scenarios of penetration testing

TEXTBOOKS:

- Ec-Council, “Ethical Hacking and Countermeasures: Attack Phases”, Delmar Cengage Learning, 2016.
- Michael T. Simpson, Kent Backman, James E. Corley, “Hands-On Ethical Hacking and Network Defense”, Cengage Learning, 2012.

REFERENCE BOOKS:

- Patrick Engebretson, “The Basics of Hacking and Penetration Testing – Ethical Hacking and Penetration Testing Made Easy”, Syngress Media, Second Revised Edition, 2013.

- Jon Erickson, “Hacking: The Art of Exploitation”, No Starch Press, Second Edition, 2008.

DAYANANDA SAGAR UNIVERSITY

Course Name: ADVANCE DATABASE(NOSQL)

Credits: 4

Course Description:

This course introduces the fundamentals and applications of NoSQL databases, focusing on modern data storage solutions. Learners will explore various data models, including key-value, document, column-family, and graph databases. Topics include data distribution techniques like sharding and replication, consistency mechanisms, and MapReduce processing. Learners will study practical use cases, such as e-commerce, event logging, recommendation systems, and emerging vector databases. By the course's end, learners will learn to design scalable and distributed database solutions while addressing challenges like performance, flexibility, and consistency in handling large-scale, unstructured data.

Course Objectives:

- Discuss the fundamental concepts and need for NoSQL databases
- Learn about NoSQL data models, including key-value, document, column-family, and graph databases
- Explore distribution models and consistency challenges in NoSQL systems
- Develop proficiency in MapReduce and data processing techniques in NoSQL databases
- Apply NoSQL concepts to real-world use cases and hybrid data management systems

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the fundamentals of NoSQL databases and their need in modern applications
- Interpret various NoSQL data models and their advanced features
- Apply distribution techniques and consistency models in database design
- Design solutions for use cases like event logging, e-commerce, and recommendation engines

- Explain vector databases, their characteristics, and applications in advanced data management

Unit 1	NoSQL Needs: The value of relational databases, Persistent data, Concurrency, Integration, Standard model, Impedance mismatch
Unit 2	Emergence and Data Models: Application and Integration databases, Clusters, Emergence of NoSQL, Aggregate data models, Key-Value and Document models
Unit 3	Advanced Data Models: Column-family stores, Relationships, Graph databases, Schemaless databases, Materialized views
Unit 4	Distribution Techniques: Single server, Sharding, Master-slave replication, Peer-to-peer replication, Combining sharding and replication
Unit 5	Consistency and Durability: Consistency (update/read consistency), Relaxing consistency, CAP theorem, Relaxing durability, Quorums
Unit 6	Versioning and Transactions: Version stamps, Business and system transactions, Version stamps on multiple nodes
Unit 7	Map-Reduce Concepts: Basic Map-Reduce, Partitioning, Combining, Composing calculations, Incremental Map-Reduce
Unit 8	Key-Value Databases Overview: What is a key-value store, Features (consistency, transactions), Query features, Scaling, Use cases
Unit 9	Use Cases and Limitations: Storing session information, User profiles, Preferences, Shopping cart data, Limitations of key-value stores
Unit 10	Document Database Basics: What is a document database, Features (consistency, transactions, availability, queries)
Unit 11	Use Cases and Applications: Event logging, Content management systems, Web analytics, E-commerce applications, Complex transactions
Unit 12	Graph Database Basics: What is a graph database?, Features (consistency, transactions, availability, queries), Scaling
Unit 13	Use Cases for Graph Databases: Connected data, Routing, Dispatch, Location-based services, Recommendation engines
Unit 14	Introduction to Vector Databases: Overview of vector databases, Key characteristics, Usage and applications

TEXTBOOKS:

- Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012.

REFERENCE BOOKS:

- Dan Sullivan, "NoSQL For Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN13: 978-9332557338)

- Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978-9351192022)
- Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage", 2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694)

Course Name: ARTIFICIAL INTELLIGENCE

Credits: 4

Course Description:

This course introduces learners to artificial intelligence concepts, focusing on problem-solving, logical reasoning, and intelligent agent frameworks. Topics include search algorithms (BFS, DFS, A*), knowledge representation, inference techniques, and natural language processing (NLP). Learners will learn about robotics fundamentals, expert systems, and machine learning concepts, such as supervised and unsupervised learning, neural networks, and deep learning techniques. The course integrates practical applications, including object detection and decision-making systems. By the end, learners will gain a foundational understanding of AI techniques, enabling them to develop intelligent solutions for real-world problems.

Course Objectives:

- Discuss the basics of AI, including intelligent agents and problem-solving strategies.
- Explore knowledge-based agents and various inference techniques
- Learn about planning, handling uncertainty, and the basics of robotics in AI
- Develop skills in natural language processing and expert systems
- Gain a foundational understanding of machine learning, neural networks, and deep learning concepts

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain intelligent agent frameworks
- Discuss problem-solving techniques
- Apply game-playing and CSP techniques
- Perform logical reasoning
- Perform probabilistic reasoning under uncertainty

Unit 1	Basics of AI: What is AI?, Intelligent agents: agents and environment, Rationality, Nature of environment, Structure of agents
Unit 2	Problem-Solving Agents: Problem-solving agents, Uninformed search strategies (DFS, BFS)
Unit 3	Informed and Adversarial Search: Informed search (Best First Search, A*, AO*), Adversarial search & games (Minimax search, Alpha-Beta pruning)
Unit 4	Knowledge-Based Agents: Knowledge-based agents, The Wumpus world example, Logic (propositional and predicate)
Unit 5	Inference Techniques: Unification, Forward chaining, Backward chaining, Resolution, Truth maintenance systems
Unit 6	Knowledge in Learning: What is learning?, Types of learning (Rote, Problem-solving, Examples), Winston's learning program, Decision Trees
Unit 7	Introduction to Planning: Blocks World problem, STRIPS (Planning)
Unit 8	Handling Uncertainty: Non-monotonic reasoning, Probabilistic reasoning, Fuzzy logic
Unit 9	Basics of Robotics: Fundamentals of Robotics, Robot Kinematics
Unit 10	Computer Vision: Introduction to image processing and classification, Object detection
Unit 11	Natural Language Processing (NLP): Introduction to NLP, Syntactic processing, Semantic analysis, Discourse and Pragmatic processing
Unit 12	Expert Systems: Architecture and role of expert systems, Case studies of expert systems
Unit 13	Introduction to Machine Learning: Supervised, Unsupervised, Reinforcement learning
Unit 14	Neural Networks and Deep Learning: Basics of ANN, Deep Learning with CNN, RNN, LSTM, Applications

TEXTBOOKS:

- Stuart Russell and Peter Norvig, "Artificial Intelligence – A Modern Approach", Fourth Edition, Pearson Education, 2021.

REFERENCE BOOKS:

- Dan W. Patterson, "Introduction to AI and ES", Pearson Education,2007.
- Kevin Night, Elaine Rich, and Nair B., "Artificial Intelligence", McGraw Hill, 2008.
- Patrick H. Winston, "Artificial Intelligence", Third Edition, Pearson Education, 2006.
- Deepak Khemani, "Artificial Intelligence", Tata McGraw Hill Education, 2013.

Course Name: ANIMATIONS

Credits: 4

Course Description:

This course introduces learners to animation tools, techniques, and principles for creating interactive and engaging animations. Topics include shape manipulation, keyframe animation, tweening, and advanced motion effects such as masking and gradients. Learners will explore scene management, timing, staging, character animation, and workspace customisation for streamlined project workflows. The course emphasises hands-on projects, focusing on designing dynamic backgrounds, UI components, and animated scenes. By the end, learners will gain practical experience in developing professional-quality animations for multimedia, entertainment, and interactive design applications.

Course Objectives:

- Discuss the animation tools, interfaces, and basic principles of animation
- Learn drawing, shape manipulation, keyframe animation, and tweening techniques
- Explore scene management, staging, timing, and workspace customisation for animation projects
- Apply character animation, motion effects, and advanced techniques for creating complex movements
- Develop practical animation projects, integrating tools, principles, and ethical standards

Course Outcomes:

At the end of the course, the learner will be able to:

- Demonstrate proficiency in using animation tools, drawing shapes, and managing frames and stages
- Apply keyframe animation, tweening techniques, and advanced motion effects like masking and gradients
- Create and manage animated scenes, ensuring proper staging, timing, and workspace customisation

- Develop character animations and integrate complex movements for dynamic projects
- Design and deliver practical animation projects, showcasing real-world applications and ethical practices

Unit 1	Introduction to Tools:Introduction to animation tools and interface, Categories of animation tools, Purpose and significance
Unit 2	Drawing and Shape Manipulation:Drawing for animation, Shape manipulation, Working with strokes and fills, Grouping shapes
Unit 3	Frame and Stage Settings:Frames per second, Stage size, Background color, Keyframes
Unit 4	Motion and Shape Tween:Introduction to tweening, Motion and shape tweening
Unit 5	Advanced Tween Techniques:Guide paths, Path animation, Masking, Animating masks, Gradients, Effects
Unit 6	Basics of Staging and Timing:Animation staging, Timing, Static and animated background scenes
Unit 7	Scene Management:Managing scenes, Duplicating and editing scenes, Process, geera architecture, examples and case studies
Unit 8	Exporting Movies:Introduction to movie export, File management, Library management
Unit 9	Workspace Customization:Customizing the workspace, Compression techniques
Unit 10	Keyframe Animation Basics:Principles of keyframe animation, Examples and case studies, Types of keyframe animations
Unit 11	Character Animation:Basics of character animation, Stick figure cycles
Unit 12	Creating Animated Scenes:Creating and managing scenes for animations, Process, Ethical standards
Unit 13	Advanced Animation Techniques:Integrating complex character and object movements
Unit 14	Practical Animation Projects:Real-world animation project examples and practical application

TEXTBOOKS:

- Russell Chun, Adobe Animate CC : Classroom in a Book - The Official Training Workbook from Adobe – 2017.
- Wayne Gilbert, Simple drawing for planning animation, 2014.

REFERENCE BOOKS:

- Preston Blair, Cartoon Animation, Walter Foster Publishing, 3 November 2020.
- Eadweard Muybridge, The Human Figure In Motion, Dover Publications, Feb 2007.

DAYANANDA SAGAR UNIVERSITY

Course Name: SOFT COMPUTING

Credits: 4

Course Description:

This course introduces soft computing techniques, including fuzzy logic, neural networks, and evolutionary algorithms, to solve complex, real-world problems. Learners will explore fuzzy sets, decision-making processes, and genetic algorithms for optimisation. Topics include hybrid systems integrating fuzzy logic and neural networks and soft computing applications in data clustering, classification, and control systems. Through case studies and hands-on examples, learners will understand the intelligent behaviour of systems. By the end of the course, learners will be able to design and apply soft computing techniques to address challenges in diverse domains effectively.

Course Objectives:

- Outline the ideas of fuzzy sets, fuzzy logic and the use of heuristics based on human experience
- Discuss the mathematical background for carrying out the optimisation associated with neural network learning
- Explore the various evolutionary Algorithms
- Familiarise with neural networks
- Illustrate the Intelligent behaviour of programs based on soft computing

Course Outcomes:

At the end of the course, the learner will be able to:

- Discuss the fundamentals of fuzzy logic operators and inference mechanisms
- Describe neural network architecture for AI applications such as classification and clustering
- Learn the functionality of Genetic Algorithms in Optimization problems
- Use hybrid techniques involving Neural networks and Fuzzy logic
- Apply soft computing techniques in real-world applications

Unit 1	NoSQL Needs: The value of relational databases, Persistent data, Concurrency, Integration, Standard model, Impedance mismatch
Unit 2	Emergence and Data Models: Application and Integration databases, Clusters, Emergence of NoSQL, Aggregate data models, Key-Value and Document models
Unit 3	Advanced Data Models: Column-family stores, Relationships, Graph databases, Schemaless databases, Materialized views
Unit 4	Distribution Techniques: Single server, Sharding, Master-slave replication, Peer-to-peer replication, Combining sharding and replication
Unit 5	Consistency and Durability: Consistency (update/read consistency), Relaxing consistency, CAP theorem, Relaxing durability, Quorums
Unit 6	Versioning and Transactions: Version stamps, Business and system transactions, Version stamps on multiple nodes
Unit 7	Map-Reduce Concepts: Basic Map-Reduce, Partitioning, Combining, Composing calculations, Incremental Map-Reduce
Unit 8	Key-Value Databases Overview: What is a key-value store, Features (consistency, transactions), Query features, Scaling, Use cases
Unit 9	Use Cases and Limitations: Storing session information, User profiles, Preferences, Shopping cart data, Limitations of key-value stores
Unit 10	Document Database Basics: What is a document database, Features (consistency, transactions, availability, queries)
Unit 11	Use Cases and Applications: Event logging, Content management systems, Web analytics, E-commerce applications, Complex transactions
Unit 12	Graph Database Basics: What is a graph database?, Features (consistency, transactions, availability, queries), Scaling
Unit 13	Use Cases for Graph Databases: Connected data, Routing, Dispatch, Location-based services, Recommendation engines
Unit 14	Introduction to Vector Databases: Overview of vector databases, Key characteristics, Usage and applications

TEXTBOOKS:

- Sajang, J.-S. R., SUN, C.-T., & MIZUTANI, E.Neuro-fuzzy and soft computing: A computational approach to learning and machine intelligence. Upper Saddle River, NJ, Prentice Hall,1997.
- Himanshu Singh, Yunis Ahmad Lone, Deep Neuro-Fuzzy Systems with Python 3. With Case Studies and Applications from the Industry, Apress, 2020.

REFERENCE BOOKS:

- Roj Kaushik and Sunita Tiwari, Soft Computing-Fundamentals Techniques and Applications, 1st Edition, McGraw Hill, 2018.
- S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI, 2003.
- Samir Roy, Udit Chakraborthy, Introduction to Soft Computing, Neuro Fuzzy and Genetic Algorithms, Pearson Education, 2013.
- S.N. Sivanandam, S.N. Deepa, Principles of Soft Computing, Third Edition, Wiley India Pvt Ltd, 2019.

Course Name: CLOUD COMPUTING

Credits: 4

Course Description:

This course comprehensively overviews cloud computing concepts, models, and applications. Learners will learn about cloud architecture, virtualisation techniques, and resource management strategies. Topics include cloud migration approaches, security challenges, and infrastructure design using platforms like IaaS, PaaS, and SaaS. The course explores storage and network virtualisation, APIs, and cloud-based solutions for real-world use cases. Learners will also study ethical standards and security management techniques to address risks in cloud environments. By the end, learners can design, deploy, and manage secure cloud-based applications effectively.

Course Objectives:

- Discuss the fundamentals of cloud computing, its models, and types
- Explore cloud architecture, virtualisation techniques, and associated challenges
- Learn cloud computing techniques, resource management, and migration approaches
- Explain the cloud security concepts, challenges, and ethical standards
- Analyse cloud infrastructure design, virtualisation management, and security architectures

Course Outcomes:

At the end of the course, the learner will be able to:

- Explain cloud computing models, types, and benefits while understanding the associated challenges
- Demonstrate virtualisation techniques, including storage, network, and hardware optimisation
- Apply cloud migration approaches and resource management strategies in cloud environments
- Develop cloud-based solutions using web technologies, APIs, and platforms

- Analyse and implement cloud security measures, including risk management, security architecture, and data protection

Unit 1	Cloud Computing Basics:Basics of cloud computing, Cloud Computing Models (PaaS, SaaS, IaaS)
Unit 2	Types of Cloud:Public, Private, Community, and Hybrid Clouds
Unit3	Cloud Architecture and Challenges:Benefits and risks, Challenges, NIST Cloud Reference Architecture, Cloud architecture and virtualization
Unit 4	Cloud Computing Techniques:Overview of cloud techniques (Grid, Utility, Fog, Edge Computing)
Unit5	Cloud Security Overview:Introduction to cloud security, Significance, Ethical Standards, Types of cloud security
Unit6	Virtualization Basics:Virtual Machines (Process, System), Emulation, Interpretation, Binary Translation, Taxonomy of Virtual Machines
Unit7	Virtualization Management:Management of virtualization, Hardware maximization, Virtualization architectures
Unit8	Storage and Network Virtualization:Storage and Network Virtualization, Horizontal and Vertical Scaling
Unit9	Accessing the Cloud:Web access technologies (SOAP, REST), Platforms, Web application frameworks, Hosting, APIs, Web browsers
Unit10	Migrating to the Cloud:Approaches to cloud migration, Seven-step model for cloud migration
Unit11	Cloud Infrastructure Design:Design of compute and storage clouds, Inter-cloud resource management, Resource provisioning, Platform deployment
Unit12	Resource Management and Exchange:Global exchange of cloud resources, Resource management
Unit13	Cloud Security Challenges:Security overview, Cloud security challenges, SaaS security
Unit14	Security Management and Architecture:Security governance, Risk management, Security monitoring, Security architecture design, Data/Application security, VM security

TEXTBOOKS:

- Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, Tata McGraw Hill, 2013.
- Dr. Kumar Saurabh, Cloud Computing, Wiley Publications, 2012.

REFERENCE BOOKS:

- Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
- John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010.
- Jim Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.

5.3 Duration of the programme

Programme	Level	Duration	Maximum duration for completion	Credits
BCA	Bachelor's Degree	3 years	(3 x 2) years (As per UGC Notification on Specification of Degree, 2014)	128 Credits

5.4 Faculty and support staff requirement

Academic Staff	Number available to meet the required delivery norms
Programme Coordinator	1 member
Course Coordinator	1 member
Course Mentor	1 member per batch of 250 students

5.5 Instructional delivery mechanisms

The instructional delivery mechanisms for the online BCA programme from CDOE, DSU, have been designed to ensure learners' engaging and practical learning experience. The CDOE has a team of qualified and experienced faculty and staff for the programme. The proficiency of the faculty team ensures that programs are thoughtfully designed and executed to meet high academic standards. This commitment to quality provides learners with a seamless, engaging, and enriching learning experience tailored to online education's unique needs and challenges. CDOE creates an environment that supports academic excellence and professional growth, empowering learners to succeed in a flexible and accessible educational setting.

CDOE will have an academic calendar outlining essential dates for significant events in the semester. This academic calendar will be shared with students at the start of each semester, ensuring clear communication and effective planning for all academic activities.

In addition to providing content through Self-Learning Materials (SLMs), learners will have access to a wide range of e-learning resources, including audio and video content, to enhance their understanding of the course material. To ensure learner engagement in the programme, various activities will be organised in the form of the following:

- **Webinars and Online Lectures:** Live sessions will be conducted to allow learners to learn directly from faculty members and engage in real-time discussions. These sessions will cover key topics and provide clarity on different concepts.

- **Discussion Forums:** Learners will be encouraged to actively participate in discussion forums designed to stimulate critical thinking and foster open communication. These forums will provide a platform for learners to share their perspectives and express their ideas without hesitation. Such forums encourage participation from all learners and provide an opportunity for discussion and gaining insights while maintaining professional decorum.

Thus, learners remain actively involved in the learning process, fostering a collaborative and enriching educational experience that aligns with the objectives of online learning.

- **Assessments:** Continuous Internal Assessments are conducted to support ongoing learning and development. Self-assessment questions included in the e-SLMs and quizzes available on the LMS provide regular opportunities for learners to conduct periodic evaluations. These quizzes can be taken multiple times, allowing learners to refine their understanding and work toward achieving correct answers. This iterative process promotes a deeper understanding of key concepts and strengthens learning outcomes. The flexibility of this approach encourages active participation, helping learners identify and address knowledge gaps while building confidence in applying their knowledge effectively. By regularly monitoring progress, learners can engage more thoroughly with the course material, ensuring continuous improvement and mastery of the subject matter.

The practical/hands-on demonstrations in the online BCA program aim to bridge the gap between theoretical knowledge and real-world application. Learners actively engage in programming tasks, system design, database management, and software development projects,

providing opportunities to apply concepts in practical settings. These hands-on sessions foster a deeper understanding of technical skills, encourage experimentation, and enhance problem-solving abilities. By simulating real-world scenarios, learners gain confidence in navigating professional challenges and build industry-relevant competencies. This practical approach ensures learners are well-equipped to tackle complex technical tasks, preparing them for success in dynamic IT roles.

Learners will be provided access to national portals such as SWAYAM and NPTEL, along with the University's digital library, which will be integrated into the LMS for supplementary reading material. This allows learners to explore additional resources beyond the prescribed syllabus. Such access will encourage learners to complement the core curriculum but also support lifelong learning, empowering learners to stay updated with the latest developments in their field of study.

5.6 Media resources - print, audio or video, online, computer-aided:

Students will be getting access to a wide range of e-learning materials, including audio and video content, faculty-led video sessions, virtual classrooms and discussion boards through the LMS. This will enable learners to track their progress in real-time through a personalised dashboard, allowing them to monitor their learning journey.

Students will also be informed about upcoming academic events. Regular notifications will be sent to remind learners about upcoming webinars, virtual classes, assignments, and discussion forums. Such notifications will help learners to manage their schedules and academic responsibilities.

Additionally, the LMS will facilitate direct communication between learners and Course Coordinators/Mentors. Learners will be able to raise queries, seek clarification, and receive responses from faculty members. This will foster a supportive learning environment and ensure that learners have the necessary guidance and resources to succeed in their studies. The LMS will be a platform to maintain an interactive and engaging online learning experience, enabling learners to actively participate in their education while receiving the support they need.

5.7 Student Support Services

Student Support services of the CDOE, DSU will be providing pre-admission learner support services like counselling about the programme including curriculum design, mode of delivery,

fee structure and evaluation methods. Post-admission learner support services include guiding learners towards accessing LMS portal, Academic Calendar and academic delivery. The support services team shall provide support/training in attending the online proctored semester end examination. The support team shall answer to the queries pertaining to conduct of end-semester examinations, evaluation and issue of certificates.

6. Procedure for Admission, Curriculum Transaction and Evaluation

The purpose of online education by CDOE, DSU is to provide flexible learning opportunities to learners to attain qualification, wherever learners are not able to attend the regular classroom teaching. Academic programmes offered for such candidates under Online Learning mode will be conducted by CDOE, DSU. The programmes/courses is termed Online mode for award of Degree.

Eligibility criteria, programme/course structure, curriculum, evaluation criteria and duration of programme shall be approved by Board of Studies and Academic Council which are based on UGC guidelines.

Candidates seeking admission in any programme offered by CDOE, DSU shall fill up the online application form available on the website. Before applying, candidates must check eligibility criteria for the programme. Details about eligibility criteria, programme structure, curriculum, duration, and fee structure are available on the University website.

6.1. Procedure for Admission

6.1.1 Minimum Eligibility Criteria for admission

- Admission to First Year Bachelor of Computer Applications shall be open to candidates who have passed the second year Pre-University or XII standard or equivalent examination recognized by the University.
- A candidate who has passed 10+2, or equivalent with a minimum of 45% marks in aggregate (40% in case of candidate belonging to SC/ST & OBC category).
- Prior to applying for admissions, candidates are advised to go through the details provided on the University website & the Programme prospectus.

Important Instructions:

- Admission granted by the University to the Programme shall be confirmed only for the candidates who fulfil the Admission Eligibility requirement by submitting all the requisite documents and has paid the semester fees.

- All other Admissions granted by the University to the Programme shall be Provisional until the candidate meet the eligibility criteria

- Provisional Admission shall stand cancelled if the candidate does not fulfil Programme eligible criteria within the stipulated time given by the CDOE, DSU.

- The University has the right to make necessary changes from time to time as deemed fit in Eligibility criteria, programme/course structure, curriculum, duration, fee structure and programme announcement dates. All changes will be notified on the website.

- Candidates should carefully read all instructions given in Programme prospectus before start of application form.

6.1.2. Fee Structure and Financial assistance policy

Suggested Fee for online BCA programme is INR 1,20,000/- (One Lakh Twenty Thousand only). Overseas learners need to remit the programme fees equivalent in USD to the University.

A scholarship of up to 10% on tuition fees will be provided to Merit learners and to learners who belong different special categories as defined in the University Policy.

6.2. Curriculum Transactions

6.2.1. Programme Delivery

DSU utilises modern technology to deliver online programs, ensuring learners receive a high standard of education. The faculty at DSU is dedicated to providing expert guidance that promotes the overall development of learners. They do more than facilitate learning—they serve as mentors, fostering an engaging environment that enhances learner retention and academic growth. The programme is designed with the goal of equipping learners with specialised expertise, helping them excel in their chosen fields. Some of the important features are:

- Online academic delivery, ensuring flexibility and accessibility for all learners.

- Regular updates and reviews of the curriculum and study materials to keep content current

and relevant.

- Live, interactive lectures conducted by CDOE, DSU faculty members and course coordinators, ensuring engagement with learners to support them in their learning journey.
- Continuous academic and technical support to assist learners throughout in their online learning journey.
- Guidance and mentoring from Course coordinators to help learners to navigate any academic challenges.
- Dedicated learning and delivery support from Course mentors.

This approach guarantees a comprehensive and supportive learning experience, where learners can focus on their academic outcomes for better professional outcomes. Through these well-structured delivery methods, DSU ensures that each learner receives the tools and guidance they need to succeed in their studies and future careers.

6.2.2. Norms for Delivery of Courses in Online Mode

Sl. No.	Credit value of the course	No. of Weeks	No. of Interactive Sessions		Hours of Study Material		Self- Study hours including Assessment etc.	Total Hours of Study (based on 30 hours per credit)
			Synchronouss Online Counselling/ Webinars/ Interactive Live Lectures (1 hour per week)	Discussion Forum/ asynchronous Mentoring(2 hours per week)	e-Tutorial in hours	e-Content hours		
1.	1 Credit	3 weeks	3 hours	6 hours	5	5	11	30
2.	2 Credits	6 weeks	6 hours	12 hours	10	10	22	60
3.	3 credits	9 weeks	9 hours	18 hours	15	15	33	90
4.	4 Credits	12 weeks	12 hours	24 hours	20	20	44	120

6.2.3. Learning Management System to support Online mode of Course delivery

The LMS platform for the online BCA programme has been specifically designed to help learners maximise their potential in their chosen field. It offers a secure and reliable learning environment that is accessible on both web and mobile devices, ensuring a consistent and seamless experience. With a user-friendly interface, the platform makes it easy for instructors

to design courses, create content, and grade assignments efficiently. Its responsive design delivers an excellent mobile experience, allowing learners to access course materials anytime, anywhere.

The LMS platform prioritises accessibility, ensuring all tools are standards-compliant and easy to navigate, including support for assistive technologies. This ensures an inclusive learning environment for all learners at all times such that learners have the flexibility to study at their own pace and on their own schedule. The availability of LMS for the online BCA programme encourages learners to develop a self-directed approach to learning in the programme.

6.2.4. Course Design

The course content has been carefully designed in accordance with the SWAYAM guidelines, employing the 4-quadrant approach to ensure a seamless and engaging learning experience. This structured approach includes four key components, each crafted to support various aspects of learner learning and engagement:

(a) Quadrant-I i.e. e-Tutorial, that contains – Faculty led Video and Audio Contents. These provide visual and auditory explanations of key concepts, offering clear and comprehensive coverage of course topics. The use of video content enhances understanding and helps learners grasp complex subjects more easily. Simulations, video demonstrations, Virtual Labs etc.

(b) Quadrant-II i.e. e-Content to contain illustrations, video demonstrations, documents as required. Curated reading resources, such as articles, case studies, and textbooks, allow learners to explore topics in greater detail. These materials complement video lectures and encourage deeper exploration of the subject matter.

(c) Quadrant-III i.e. Discussion forums to raise and clarify doubts on real time basis bythe Course Coordinator(s) and their team. Interactive online discussion platforms enable learners to engage with their peers and instructors. These forums foster collaboration, allowing learners to share insights, ask questions, and discuss ideas, creating a rich, supportive learning environment.

(d) Quadrant-IV i.e. Self-Assessment, that contains MCQs, Problems, Quizzes,Assignments with solutions and Discussion forum topics. Quizzes, assignments, and tests are integrated throughout the course to help learners gauge their understanding and track their progress. These self-assessment tools encourage active learning and allow learners to identify areas for

improvement.

By utilising the 4-quadrant approach, the course content ensures a balanced and holistic learning experience that promotes both theoretical knowledge and practical application.

6.2.5. Academic Calendar

The Academic Calendar indicates the timelines for the different academic activity for the Programme in the semester:

Sl. No.	Week	Event(s)
1	Week 1	Induction to the Programme.
2	Week 2-14	Commencement of live lecture sessions.
3	Week 3	Opening of Continuous Internal Assessments (CIA).
4	Week 4	Internal Assignment(s) submission.
5	Week 2-11	Discussion Forums.
6	Week 12	Closure of Internal Assignment(s) submissions.
7	Week 13	Semester End Examination (SEE) - Time Table.
8	Week 13-14	Semester End Examination (SEE) - Registration.
9	Week 15	Exam Admit Card download.
10	Week 16 onwards	Semester End Examination (SEE).
11	Week 17 onwards	Registration for next higher semester.

6.3 Evaluation

Every learner shall be assessed for a course through Continuous Internal Assessment (CIA) and Semester End Examination (SEE) as prescribed. CIA and SEE shall respectively have 30:70 percent weightage.

Continuous Internal Assessment (CIA) for Theory Courses shall be conducted for 30 marks in the form of assignments. Semester End Examination Assessment (SEE) for Theory Courses shall be conducted for 70 marks. The SEE question paper shall comprise of objective and descriptive type questions. The SEE will be conducted with technology support as a remote proctored examination.

Continuous Internal Assessment (CIA) for Practical Courses shall be conducted for 30 marks. Semester End Examination (SEE) for Practical/ Term Paper/ Mini Project/ Project shall be conducted for 70 marks. Additional details shall be made available in the respective Practical

Guidelines Manual. Learners will have to submit their internal assignments and attend viva-voce.

A learner's performance in a course shall be judged by taking into account the results of CIA and SEE together. A learner has to obtain and satisfy the following conditions to be declared as pass in each course:

(i) minimum 40% of marks in CIA
(ii) minimum 40% of marks in SEE
(iii)minimum 40% of marks in aggregate considering both CIA & SEE

- Students must score minimum 40% marks for project-based courses.
- There shall be no improvement of Continuous Internal Assessment marks if they are above 40%.
- If a student fails in any one component (failure to get 40% marks either in CIA or SEE), then the student will be required to re-appear for that component only (CIA or SEE as the case may be).
- There shall be no improvement of Semester End Examination marks if they are above 40%.

DSU shall be complying as per the prevailing regulatory directions on the conduct of the examinations.

6.3.1 Question Paper Pattern

The Question Paper for the Semester End Examination Assessment (SEE) for Theory Courses shall be conducted for 70 marks shall comprise of three sections:

- Section – A for 20 marks comprising TEN Multiple Choice Questions (MCQ) of 2 mark each.
- Section – B for 30 marks of Short Answer type Descriptive Questions of 6 marks each for which a learner shall be need to write answers for 5 out of 6 questions.
- Section – C for 20 marks of Long Answer type Descriptive Questions of 10 marks each for which a learner shall be need to write answers for 2 out of 3 questions.

Section A (Answer ALL)	**Questions x Marks**	**Marks**
Ten Multiple Choice Questions	10 x 2	20
Section B – Answer **Five** out of **Six** questions		
Descriptive Questions (Short Answers)	5 x 6	30
Section C – Answer **Two** out of **Three** questions		
Descriptive Questions (Long Answers)	2 x 10	20
Total		**70**

6.3.2 Distribution of Marks in Continuous Internal Assessments

The following procedure shall be followed for awarding internal marks for courses. Learner must submit two assignments each carrying 30 marks and average of both will be considered as internal assessment marks.

6.3.3 Passing Minimum

The learners are considered as passed in a course if they score 40% marks in the Continuous Internal Evaluation (CIA) and Semester-End Examinations (SEE) individually. If a learner fails in any one component (failure to get 40% marks either in CIA or SEE), then the learner will be required to re- appear for that component only.

6.3.4 Marks and Grades

Marks shall be awarded for both CIA and SEE. The grading will normally be based on CIA and SEE. Relationships among Grades, Grade points and % of marks are listed as per the below mentioned criteria, where F is Fail and IC is ABSENT:

GRADE	GRADE POINTS	DESCRIPTION	% MARKS
O	10	Outstanding	90 to 100
A+	9	Excellent	80 to 89
A	8	Very Good	70 to 79
B+	7	Good	60 to 69

B	6	Above Average	55 to 59
C	5	Average	50 to 54
P	4	Pass	40 to 49
F	0	Fail	< 40
IC	-	In Complete	-

Class Equivalence of Grade points

CGPA	Class/Division
>=5.0 - < 5.75	Pass Class
>=5.75 -<6.75	Second Class
>=6.75 - <7.75	First Class
>=7.75 – 10	First Class with Distinction

For a semester:

The SGPA is calculated on the basis of grades obtained in all courses, except audit courses and courses in which F grade or below, registered for in the particular semester.

$$\text{SGPA} = \frac{\text{Points secured in the semester (O} - \text{P Grades)}}{\text{Credits registered in the semester, excluding audit courses}}$$

For the entire programme:

The CGPA is calculated on the basis of all pass grades, except audit courses.

$$\text{CGPA} = \frac{\text{Cumulative points secured in all the passed courses (O} - \text{P Grades)}}{}$$

Cumulative registered credits, excluding audit Courses

7. Requirement of the Laboratory Support and Library Resources

7.1 Laboratory Support

No lab based courses are offered in this program.

7.2 Library Resources

CDOE, DSU provides an exceptional library facility to support the academic needs of learners enrolled in the online BCA program. The Central Library at DSU is well-stocked with a vast array of reference books, including key titles relevant to the online BCA curriculum.

For learners enrolled in the online mode of education, the University offers digital library access, which provides an extensive collection of e-books, journals, and academic databases. This digital resource ensures that online learners have equal access to critical academic content, supporting them in their coursework and research. To further enhance the academic experience, DSU will provide access to educational platforms like SWAYAM, Scopus, and Knimbus, to the learners. These memberships provide access to an extensive range of academic content, including journals, articles, and research papers, enhancing the research capabilities among learners.

To ensure a holistic learning experience, DSU has integrated e-learning resources into the LMS for the online BCA program. The LMS serves as a centralised hub for all course materials, including e-books, articles, and other resources. Additionally, e-tutorial lectures are made available, offering learners the flexibility to study at their own pace while reinforcing key concepts covered in their courses. This seamless integration of resources ensures that learners have everything they need to succeed in their academic journey, regardless of their mode of study.

Beyond traditional academic resources, DSU will organise webinars and other virtual interactions by industry professionals for learners. Such events will provide learners with valuable insights into contemporary trends, challenges, and best practices in the business world. Through a combination of comprehensive library resources, digital access, and expert industry engagement, the CDOE at DSU will provide online BCA learners to strive for academic

excellence.

8. Cost Estimate of the Programme and the Provisions

The costs towards the programme study material development and academic delivery system depend on the total programme credits and the number of learners. DSU, known for academic excellence, has always complied with the UGC regulations. The programme related expenses towards e-content and IT infrastructure setup cost 50% of the programme fee revenue and the balance for the academic delivery of the programme which depends on the learner strength.

9. Quality assurance mechanism and expected programme outcomes

The quality of the online BCA programme is defined by the professionalism of its curriculum, meticulously designed to meet the demands of the evolving IT and computing fields. A well-structured syllabus, combined with dedicated efforts and effective course delivery, is essential to the program's success. The primary objective of the online BCA programme is to provide learners with a strong foundation in computing principles and practical skills in software development, database management, and web technologies. Additionally, expertise in information and communication technology (ICT) gained through the programme enhances career opportunities, enabling learners to advance in the workplace and contribute significantly to the IT industry and society at large.

The effectiveness of the programme will be measured through various benchmarks, including learners' performance in their final semester examinations and the quality of their hands-on project work. These evaluations will offer valuable insights into the programme's impact on learners' technical skill acquisition and practical knowledge. Furthermore, continuous feedback from key stakeholders, including learners, faculty, and industry representatives, will play a vital role in maintaining and improving the quality of the programme. By analysing and implementing this feedback, the programme can be refined to better align with the evolving needs of the IT sector.

CDOE, DSU has established the Centre for Internal Quality Assurance (CIQA) to support the Director, CDOE, DSU, to conduct periodic review and assessments, ensuring the implementation of quality measures and effective programme delivery. CIQA plays an integral

role in evaluating materials prepared by CDOE, including syllabi, self-learning materials (SLMs), and e-learning content. It also conducts studies to assess the effectiveness of the learning methods adopted. Moving forward, CIQA will engage in benchmarking academic delivery quality, performing various analyses, and guiding stakeholders to consistently enhance programme standards.

The Centre for Internal Quality Assurance Committee (CIQAC), chaired by the Vice Chancellor and comprising internal and external experts, oversees CIQA's operations. The committee approves reports generated by CIQA on the effectiveness of quality assurance systems and processes.

In alignment with the National Assessment and Accreditation Council (NAAC) guidelines, DSU has also established the Internal Quality Assurance Cell (IQAC). This body includes academicians, industry representatives, and other stakeholders working collectively to achieve the goals of quality enhancement and sustainability. As a continuous process, IQAC's primary task is to develop and implement systems for conscious, consistent, and catalytic improvements in the institution's overall performance. IQAC promotes institutional functioning geared toward quality enhancement by fostering a culture of quality and institutionalising best practices.

Guided by the quality monitoring mechanisms prescribed by the University Grants Commission (UGC), CIQA conducts institutional quality audits to promote excellence and propagate best practices in quality assurance. The University has implemented an effective system for gathering regular feedback from stakeholders, using it to refine its programmes. Regular self-assessments are conducted to improve systems, processes, and, ultimately, the quality of the BCA programme.

The expected programme outcomes of the online BCA programme equip graduates with a robust foundation in computing and problem-solving skills. They will be able to apply knowledge of mathematics, algorithmic principles, and computing fundamentals to design and model computer-based systems of varying complexity. Graduates will develop the ability to critically analyse, categorise, and solve challenges in computer science, providing effective software solutions for complex scientific and business problems while considering public health, safety, and cultural, societal, and environmental factors. With a strong understanding

of the impact of software solutions on the environment and society, they will actively contribute to sustainable development through responsible practices. Proficiency in modern tools, techniques, and skills will enable graduates to create integrated computing solutions while upholding social, cultural, and ethical responsibilities in their roles as individuals or team members with a collaborative and positive approach.

Graduates will also excel in effective communication, engaging with diverse audiences and preparing technical documents tailored to various stakeholders. They will demonstrate the ability to function as members or leaders of multidisciplinary teams, working toward common goals with professionalism and efficiency. A commitment to lifelong learning will keep them updated with emerging technologies and trends, allowing them to adapt to the rapidly changing IT landscape. Additionally, a research-oriented mindset will enable graduates to uphold scientific integrity and objectivity, contributing to advancements in computing and technology. These outcomes prepare graduates to meet the dynamic demands of the IT industry with competence, ethics, and a focus on sustainable development.

28/8/25

Registrar