SNDT Women's University, Mumbai

Post-Graduate Diploma in Computer Science and Application

(PGDCSA)

As Per NEP - 2020

Syllabus

w.e.f.

A.Y.: 2024-2025

1, Nathibai Thackersey Road, Mumbai- 400020

www.sndt.ac.in

Programme Template:

Programme Degree		Post-Graduate Diploma in Computer Science and Application (PGDCSA)
Parenthesis if any (Specialization) e.g. History, Human Development, English, etc.		NA
Preamble (Brief Introduction to the programme)		The Post-Graduate Diploma in Computer Science and Application is a rigorous academic program designed to provide advanced knowledge and skills in the field of computer science and its applications. This program is crafted to cater to the evolving needs of the industry and academia, equipping students with the expertise required to excel in a dynamic technological landscape. Through a blend of theoretical foundations and practical hands-on experience, this diploma aims to foster critical thinking, problem-solving abilities, and innovative approaches to tackle real-world challenges in computer science and its diverse applications. Recognizing the interdisciplinary nature of modern computing, the curriculum encompasses a wide range of topics including but not limited to computer programming, algorithms and data structures, database systems, software engineering principles, computer networks, artificial intelligence, machine learning, and cybersecurity. Moreover, this diploma emphasizes the importance of ethical considerations, professionalism, and lifelong learning, empowering graduates to contribute responsibly and ethically to the advancement of technology and society. With a faculty comprising experienced academics and industry professionals, state-of-the-art facilities, and a conducive learning environment, this program offers a comprehensive educational experience that prepares graduates for successful careers in various sectors such as software development, IT consulting, research, and academia. By undertaking the Post-Graduate Diploma in Computer Science and Application, students embark on a transformative journey of intellectual growth, skill development, and career advancement, poised to make significant contributions to the ever-expanding field of computer science and its applications.
Programme Specific Outcomes (POs)		**After completing this programme, Learner will**
	1.	**Demonstrate Advanced Knowledge:** Exhibit a deep understanding of theoretical principles and practical applications across various domains of computer science including programming, algorithms, data

		structures, software engineering, and artificial intelligence.
	2.	**Apply Problem-Solving Skills:** Utilize analytical and critical thinking skills to identify, formulate, and solve complex problems in computer science and its diverse applications, employing appropriate methodologies and tools.
	3.	**Design and Develop Software Solutions:** Design, develop, and implement software solutions using industry-standard practices and methodologies, considering factors such as scalability, reliability, and security.
	4.	**Analyze and Evaluate Systems:** Analyze, evaluate, and optimize computer systems and applications, identifying areas for improvement and implementing solutions to enhance performance, efficiency, and usability
	5.	**Communicate Effectively:** Communicate technical concepts, ideas, and solutions effectively through oral presentations, written reports, and documentation, catering to diverse audiences including technical and non-technical stakeholders.
	6.	**Collaborate in Interdisciplinary Teams:** Collaborate effectively in interdisciplinary teams, demonstrating leadership, teamwork, and interpersonal skills to achieve common goals and objectives in the development and implementation of computing projects.
	7.	**Adapt to Technological Advancements:** Stay abreast of emerging technologies, trends, and developments in the field of computer science, demonstrating a commitment to lifelong learning and professional development.
	8.	**Adhere to Ethical Standards:** Uphold ethical principles and professional standards in all aspects of computer science practice, demonstrating integrity, accountability, and responsibility in decision-making and conduct.
	9.	**Contribute to Society:** Contribute responsibly and ethically to the advancement of technology and society, addressing societal challenges and promoting inclusivity, diversity, and sustainability in computing initiatives.
	10.	**Pursue Continuous Improvement:** Engage in reflective practice and continuous improvement, seeking opportunities for further education, training, and skill enhancement to remain competitive and adaptable in a dynamic technological landscape.
Eligibility Criteria for Programme		A woman graduate from recognized university with aggregate marks of not less than 50% (Open Category) and 45% (Reserved Category).

Intake (For SNDT WU Departments and Conducted Colleges)		35

Structure with Course Titles

(Options related to our area of study to be provided with "OR" for baskets of different types)

Postgraduate Programme of 2 years:

SN	Courses	Type of Course	Credits	Marks	Int	Ext
	Semester I					
PG 1.1	Python Programming	Major (Core)	4	100	50	50
PG 1.2	Database Management Systems	Major (Core)	4	100	50	50
PG 1.3	Operating System	Major (Core)	4	100	50	50
PG 1.4	Fundamental of Computer Science	Major (Core)	2	50	50	0
PG 1.5	Elective-I	Major (Elective)	4	100	50	50
PG 1.6	Research Methodology	Minor Stream (RM)	4	100	50	50
			22	**550**	**300**	**250**
	Semester II					
PG 2.1	Data Structure and Analysis of Algorithm	Major (Core)	4	100	50	50
PG 2.2	Data Communication and Networking	Major (Core)	4	100	50	50
PG 2.3	Web Technology	Major (Core)	4	100	50	50
PG 2.4	Introduction to AI	Major (Core)	2	50	0	50
PG 2.5	Elective-II	Major (Elective)	4	100	50	50
PG 2.6	OJT	OJT	4	100	50	50
			22	**550**	**250**	**300**

Electives	
Elective - I	**Elective - II**
A. Cyber Security	A. Data Warehousing and Data Mining
B. Computer Oriented Statistical Techniques (Using R)	B. Introduction to Data Science

RM: Research Methodology
OJT: On-Job Training
RP: Research Project

Course Syllabus
Semester I

1.1 Major (Core)

Course Title	**Python Programming**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply Python for Data Science and statistics.
	• Analyse and implement exception handling in Python applications for error management.
	• Evaluate the utility of Python as a scripting language for developers.
	• Design and program Python applications, including object-oriented programs with Python classes.
	• Apply Python for Data Science and statistics.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply decision-making constructs, loops, and control statements in Python.
	• Analyse and implement OOP concepts, including inheritance, polymorphism, and encapsulation, along with anonymous functions.
	• Evaluate and implement exception handling using try, except, and finally blocks.
	• Design Python programs utilizing key features of the language.
Content Outline	• **Introduction to Python Programming**–Why Python? – Essential Python libraries – Python Introduction- Features, Identifiers, Reserved words, Indentation, Comments, Built-in • **Data types and their Methods:** Strings, List, Tuples, Dictionary, Set – Type Conversion- Operators • **Decision Making- Looping**- Loop Control statement- Math and Random number functions • **User defined functions** – function arguments & its types User defined Modules and Packages in Python- • **Files**: File manipulations, File and Directory related methods Python Exception Handling. • **OOPs Concepts** – Class and Objects, Constructors – Data hiding- Data Abstraction- Inheritance, Types of Inheritance, Polymorphism, Encapsulation, Anonymous Function.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply arithmetic operations, indexing, slicing, and Boolean indexing on NumPy arrays.
	• Analyse and manipulate NumPy ndarrays, including specifying data types.
	• Evaluate and implement essential functionality such as dropping entries, indexing, selection, and filtering in pandas.
	• Design data manipulation workflows using unique values, value counts, and membership in pandas DataFrames.

Content Outline	• **NumPy Basics:** Arrays and Vectorized Computation- The NumPy ndarray- Creating ndarrays- Data Types for ndarrays- Arithmetic with NumPy • **Arrays**- Basic Indexing and Slicing – Boolean Indexing Transposing Arrays Universal Functions: Fast Element-Wise Array Functions- Mathematical and Statistical Methods-Sorting Unique and Other Set Logic. • **Introduction to pandas Data Structures**: Series, Data Frame, Essential Functionality: Dropping Entries Indexing, Selection, and Filtering- Function Application and Mapping- Sorting and Ranking. • **Summarizing and Computing Descriptive Statistics**- Unique Values, Value Counts, and Membership. Reading and Writing Data in Text Format.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

1. Write a Python program that includes all the built-in data types, demonstrates type conversion and uses various operators.
2. Create a Python program that uses decision making and looping constructs to solve a problem, such as finding the factorial of a number or simulating a simple game.
3. Write a Python program with user-defined functions and demonstrate the use of different types of function arguments.
4. Create a Python module/package containing multiple functions and demonstrate its usage in a separate Python program.
5. Write a Python program that reads a text file, applies some transformations (like changing case, removing punctuation), and writes the result back to a new file.
6. Implement a class in Python demonstrating the concepts of OOPs including constructors, data hiding, abstraction, inheritance, polymorphism, and encapsulation.

Module 2:

1. Create a Python program that demonstrates the creation and manipulation of NumPy arrays, including arithmetic operations, indexing, slicing, and transposing.
2. Write a Python program using NumPy for statistical analysis, like computing the mean, median, mode, variance, and standard deviation of a set of numbers.
3. Write a Python program using pandas to create Series and DataFrames. Demonstrate dropping entries, indexing, selection, and filtering on these data structures.
4. Create a Python program that uses pandas to read a CSV file, applies some transformations (like filtering, sorting), and writes the data back to a new CSV file.

5. Write a Python program using pandas to summarize and compute descriptive statistics on a dataset. The dataset could be any publicly available dataset like the Titanic dataset or the Iris dataset.

References

1. Liang, Y. D. (2012). Introduction to Programming using Python. Pearson.
2. McKinney, W. (2018). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython (2nd ed.). O'Reilly.
3. Savaliya, M. T., Maurya, R. K., & Magar, G. M. (2021). Programming with Python. SYBGEN Learning India Pvt. Ltd.
4. VanderPlas, J. (2017). Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly.

1.1 Major (Core)

Course Title	**Python Programming (Lab)**
Course Credits	**2**
Course Outcomes	**After completion of this Course, the students will be able to**
	• Apply Python syntax, including variables, data types, control structures, and functions.
	• Analyse and implement commonly used Python libraries and modules, such as NumPy and Pandas.
	• Evaluate the effectiveness of Python libraries in solving data manipulation and analysis tasks.
	• Design Python scripts using fundamental syntax and libraries to address real-world problems.
Module1 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply decision-making constructs and loops in Python.
	• Analyze their behavior.
	• Evaluate their effectiveness and efficiency.
	• Design complex programs using these constructs.
Content Outline	• **Demonstrate a programs on Decision Making**- Looping- Loop Control statement- Math and Random number functions User defined functions, File manipulations • **Demonstrate a programs on OOPs** Concepts such as Class and Objects, Constructors – Data hiding- Data Abstraction- Inheritance, Types of Inheritance, Polymorphism, Encapsulation
Module2 (Credit 1)	
Learning	**After learning this module, learners will be able to**

Outcomes	
	• Apply NumPy to create and manipulate ndarrays with specific data types.
	• Analyze the behavior of various array operations.
	• Evaluate efficiency of arithmetic operations.
	• Design complex applications using indexing, slicing and Boolean indexing on NumPy arrays.
Content Outline	• **Demonstrate a programs on Arrays**- NumPy Basics such as Creating ndarrays data types for ndarray, Basic Indexing and Slicing • **Mathematical and Statistical Methods** and implementing pandas for implementing data sets

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

1. Web Scraper: Write a Python program using built-in data types, functions, loops, and decision-making constructs that scrapes data from a website (such as a news site or weather forecast site) and stores the data in a text file.
2. Contact Book Application: Design a Python program that implements a simple contact book. Use OOP concepts to create classes for 'Contact' and 'AddressBook'. The 'Contact' class should have data members for storing name, phone number, email, etc., and 'AddressBook' class should manage multiple 'Contact' objects.
3. File Organizer: Write a Python program that organizes files in a specified directory on your computer. The program should categorize files based on file type (e.g., .txt, .docx, .pdf) and move them to corresponding sub-folders.

Module 2:

1. Stock Market Analyzer: Use NumPy and pandas to analyze stock market data. Fetch data from a free API like Alpha Vantage, perform operations like calculating moving averages, daily return, and visualize the data using matplotlib or seaborn.
2. Survey Data Analysis: Use a dataset from an online survey tool (like Google Forms) exported as a CSV. Load the data into a pandas DataFrame, clean the data, filter it, and perform statistical analysis.
3. Health Tracker: Design a program that tracks personal health information. Users can input their daily calories, exercise, water intake, etc. Use pandas to store and manipulate this data, and NumPy to calculate weekly averages or other statistics.

References

1. Liang, Y. D. (2012). Introduction to Programming using Python. Pearson.
2. McKinney, W. (2018). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython (2nd ed.). O'Reilly.
3. Savaliya, M. T., Maurya, R. K., & Magar, G. M. (2021). Programming with Python. SYBGEN Learning India Pvt. Ltd.

4. VanderPlas, J. (2017). Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly.

1.2 Major (Core)

Course Title	**Database Management System**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply the fundamental concepts of a database system and use SQL for database management.
	• Analyze the design and efficiency of databases using E-R and Relational Models. Evaluate the effectiveness of SQL operations and the impact of different transaction and recovery techniques
	• Design complex databases emphasizing efficient data storage, retrieval, and robust transaction recovery systems.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Understand the core concepts of a database system and apply them in practical scenarios.
	• Analyze and synthesize database designs using E-R and Relational Models
	• Evaluate the effectiveness and efficiency of these models in different use cases.
	• Design complex databases using E-R and Relational Models for efficient data management.
Content Outline	• **Introduction**: Database System application, Database System versus File systems, View of Data, Data Models, Database Languages, Database Users and administrator. DBA Roles and activity, Database system structure. • **Entity-Relational Model**: Basic Concepts ,Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets, Extended E-R features, Design of E-R Database Schema, Reduction of an E-R Schema to Tables • **Relational model**: Structure of Relational Databases, Relational Algebra, Tuple Relational Calculus, Domain Relational Calculus
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply SQL commands in various scenarios and understand the use of constraints.
	• Understand and implement concepts of PL/SQL and Triggers
	• Evaluate and demonstrate the process of decomposing databases and normalizing data using normalization techniques.
	• Design and analyze database systems with a focus on efficient data retrieval using concepts of hashing and indexing.
Content Outline	• **SQL:** SQL commands , Functions, Data Constraints, Grouping Data, Sub-queries, Joins, Performance Tuning, Security Management ,PL/SQL, Triggers. • **Integrity & Security**: Domain Constraints, Referential Integrity, Assertions, Triggers, Privileges in SQL. • **Relational Database Design**: Functional Dependencies, Decomposition, Normalization 1NF-5NF,BCNF • RAID , Improvement of Reliability & Performance Indexing & Hashing – Basic Concepts, Ordered Indices, B+ & B Tree Index Files, Static & Dynamic Hashing, Comparison of Ordered Indexing & Hashing. • **Concurrency Control:** Protocols-Lock Based, Timestamp-

	based, Validation Based, Deadlock Handling & Concurrency • **Recovery System:** Failure Classification, Storage Structure, Recovery & Atomicity • **Introduction and Overview of NoSQL Database**

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write an essay discussing the core concepts of a database system and their application in practical scenarios.

Activity 2: Design a database schema for a hypothetical business scenario using the E-R model. Translate this design into the Relational Model.

Activity 3: Write a set of relational algebra expressions to perform various operations on your designed database.

Module 2:

Activity 1: Write a series of SQL commands for various operations, such as data insertion, update, deletion, and retrieval in the context of a specific business scenario. Include constraints in your commands and explain their purpose.

Activity 2: Write a PL/SQL script that includes the use of triggers. Execute the script and document the results.

Activity 3: Normalize the database schema created in Module 1 Activity 2. Write a report detailing the normalization process and its effects on database performance and integrity.

Activity 4: Explain the process of indexing and hashing in databases. Write SQL commands to create indices on specific columns in your database. Discuss the performance improvements these indices can bring.

References

1. Korth, H., Silberschatz, A., & Sudarshan, S. (2005). Database System Concepts (5th ed.). McGraw-Hill.
2. Elmasri, R., & Navathe, S. B. (2008). Fundamentals of Database Systems (3rd ed.). Pearson Education India.
3. Ramakrishnan, R., & Gehrke, J. (2002). Database Management Systems. McGraw-Hill International Edition.
4. Hoffer, J. A., Prescott, M., & McFadden, F. (2004). Modern Database Management (7th ed.). Prentice Hall.
5. Rob, P., & Coronel, C. (2004). Database Systems: Design, Implementation and Management. Thomson Publication.
6. Kroenke, D. M. (2005). Database Processing: Fundamentals, Design, Implementation (10th ed.). Prentice-Hall.
7. O'Neil, P. (2000). Database Principles Programming Performance (2nd ed.). Morgan Kaufmann Publishers, Inc.

8. Urman, S. (2002). Oracle8i PL/SQL Programming.

1.2. Major (Core)

Course Title	**Database Management System (Lab)**
Course Credits	**2**
Course Outcomes	**After completion of this Course, the students will be able to**
	• Apply and develop proficiency in database languages, particularly SQL, in different scenarios.
	• Understand and analyze SQL commands, functions, data constraints, grouping data, subqueries, joins, and performance tuning.
	• Evaluate the effectiveness and efficiency of these commands and functions in various use cases.
	• Design complex SQL queries and optimize them for better performance.
Module1 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply knowledge to implement database schemas with appropriate constraints.
	• Analyze the effects of normalization, different types of joins, and clauses on database performance and integrity.
	• Evaluate the effectiveness of these techniques in various database scenarios.
	• Design databases with optimized schemas, using normalization, joins, and different clauses to ensure efficient data management.
Content Outline	• Database • Table Creation • Defining Schema • Constraints • Normalization
Module2 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply the concepts of PL/SQL to execute procedures, functions, triggers, cursors, etc.
	• Analyze the behavior and performance of these PL/SQL elements in various scenarios.
	• Evaluate the effectiveness and efficiency of these elements in different use cases.

	• Design complex PL/SQL scripts that effectively utilize procedures, functions, triggers, and cursors for efficient database management.
Content Outline	• SQL Basic Queries -Joining, and Clause's implementation • Procedure, Function execution, PLSQL Script Execution, Stored Procedure, Function, Packages Execution • Cursor, Trigger Writing

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Design and implement a database schema for a hypothetical business scenario. The schema should include appropriate constraints to maintain data integrity.

Activity 2: Normalize the database schema created in Activity 1. Write a report detailing the normalization process and its effects on database performance and integrity.

Activity 3: Using the same business scenario, write SQL queries to demonstrate the use of different types of joins and clauses. Analyze the results and discuss their implications for data management.

Module 2:

Activity 1: Write a PL/SQL script that includes the use of procedures, functions, triggers, and cursors. Execute the script and document the results.

Activity 2: Analyze the behavior and performance of the PL/SQL elements used in your script. Discuss their effectiveness and efficiency in the context of the hypothetical business scenario.

Activity 3: Modify your PL/SQL script to improve database management efficiency. Explain the changes made and their expected impact on database management.

References:

1. Korth, H., Silberschatz, A., & Sudarshan, S. (2005). Database System Concepts (5th ed.). McGraw-Hill.
2. Elmasri, R., & Navathe, S. B. (2008). Fundamentals of Database Systems (3rd ed.). Pearson Education India.

1.3 Major (Core)

Course Title	**Operating System**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply knowledge of computer system architecture and operating systems principles in practical scenarios.
	• Analyze the impact of virtualization techniques on system performance and efficiency.
	• Evaluate processor management and memory allocation strategies to optimize system operations.
	• Design solutions using advanced file systems to ensure reliable data management.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply knowledge of computer system organization and architecture, with a focus on the role of operating systems, in diverse computing environments.
	• Analyze the structure and operations of operating systems, including process management, memory management, storage management, protection, and security.
	• Evaluate various operating system services provided to users and applications, and their effectiveness in different scenarios.
	• Understand and apply the user-operating system interface, system calls, types of system programs, and process scheduling algorithms in practical use cases.
Content Outline	• **Introduction to Operating Systems (OS):** Computer-System Organization, Computer-System Architecture, Operating-System Structure, Operating-System Operations, Process Management, Memory Management, Storage Management, Protection and Security, Distributed Systems, Special-Purpose Systems, Computing Environments. • Operating-System Services, User Operating-System Interface, System Calls, Types of System Calls, System Programs, Operating-System Design and Implementation, Operating-System Structure, Virtual Machines, Operating-System Generation. • **Processor Management:** Process concept, Process scheduling, Operations on Processes, Interprocess Communication, Multithreading models, threading issues, Process scheduling algorithms, Thread scheduling, Multiple processor Scheduling. • **Process Coordination:** Synchronization, Semaphores, Monitors, Deadlocks characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock Avoidance, Deadlock detection, recovery from deadlock.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply and understand various memory management techniques such as swapping, contiguous memory allocation, paging, and segmentation in different computing scenarios.
	• Analyze the structure of the page table and its role in memory addressing for efficient resource management.

	• Evaluate the organization and implementation of file systems, including structure, directory implementation, allocation methods, and free-space management.
	• Explore and understand the efficiency and performance considerations in file management, including recovery mechanisms and log-structured file systems.
Content Outline	• **Memory Management:** Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation • **Virtual memory Management:** Demand Paging, Copy- on-Write, Page replacement, Allocation of Frames, Thrashing. • **File Management:** • File Concept, File Access Methods, Directory Structure, File Sharing, File Protection, File-System Structure, File- System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery, Log-Structured File Systems, NFS.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write a report on the organization and architecture of computer systems, focusing specifically on the role of operating systems. Discuss the structure and operations of operating systems, including process management, memory management, storage management, protection, and security.

Activity 2: Write a program that simulates the use of system calls. Document the purpose of each system call used and discuss its importance in the operating system.

Activity 3: Simulate a process scheduling algorithm, document the steps, and evaluate its effectiveness in managing processor resources.

Module 2:

Activity 1: Write an essay discussing various memory management techniques such as swapping, contiguous memory allocation, paging, and segmentation. Include examples of scenarios where each technique would be most effective.

Activity 2: Design a page table for a hypothetical system. Discuss the role of the page table in memory addressing and resource management.

Activity 3: Write a report on the organization and implementation of file systems. Discuss the structure, directory implementation, allocation methods, and free-space management, and evaluate their efficiency and performance.

References

1. Silberscatz, A., Galvin, P. B., & Gagne, G. (2005). Operating System Concepts (7th ed.). John Wiley and Sons, Inc.
2. Milenkovic, M. (1992). Operating Systems Concepts And Design (2nd ed.). McGraw-Hill International Editions.
3. Stallings, W. (2005). Operating Systems: Internals and Design Principles (5th ed.). Prentice Hall.

4. Tanenbaum, A. (2007). Modern Operating Systems (3rd ed.). Pearson Education.

1.3 Major (Core)

Course Title	**Operating System (Lab)**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply theoretical knowledge and hands-on experience in implementing and experimenting with various operating system concepts and functionalities in real-world scenarios.
	• Analyze system issues and apply troubleshooting techniques to optimize performance for efficient system operations.
	• Evaluate the effectiveness of implemented operating system concepts and functionalities, ensuring they meet real-world scenario requirements.
	• Design strategies to optimize system performance, leveraging hands-on experience and theoretical knowledge.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply best practices to achieve successful configuration in a virtual environment, ensuring optimal system performance.
	• Analyze storage requirements and execute successful file system management for efficient storage and retrieval of data.
	• Evaluate the effectiveness of the configured virtual environment and file system management, ensuring they meet system performance and data storage needs.
	• Design strategies for optimal system performance and efficient data storage and retrieval in a virtual environment.
Content Outline	• Installation of OS on Virtual Machine (VM, Oracle BOX etc) • File Commands, Commands to Access File Contents, File Systems, System Commands
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply knowledge of scripting languages and command-line tools to efficiently implement programs on the UNIX Shell.
	• Analyze system operations to determine the most effective use of scripting languages and command-line tools in UNIX Shell programming.
	• Evaluate the effectiveness of implemented programs on the UNIX Shell, ensuring they meet system operation needs.
	• Design efficient system operations strategies by leveraging the power of UNIX Shell programming, scripting languages, and command-line tools.
Content Outline	• Shell: Introduction Section,Reading User Input, Arithmetic, Positional Parameters • Command Line Arguments, Conditional Constructs and Flow Control Section, Looping Commands

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:
Activity 1: Set up a virtual environment by installing an OS on a Virtual Machine. Write a report detailing the steps you took and any challenges you encountered.
Activity 2: Perform a series of file commands and system commands to manage a file system in the virtual environment. Document the commands used and their outcomes, explaining their importance in efficient storage and retrieval of data.

Module 2:
Activity 1: Write a simple program using a scripting language that can be run on the UNIX Shell. The program should take command line arguments and use conditional constructs and flow control.
Activity 2: Modify your program to incorporate looping commands. Document the changes made and explain how these enhance the efficiency of the program.

References:

1. Cannon, J. (2015). Shell Scripting: How to Automate Command Line Tasks Using Bash Scripting and Shell Programming.

1.4 Major (Core)

Course Title	**Fundamental of Computer science**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Acquire a comprehensive understanding of computer fundamentals, encompassing both hardware and software aspects.
	• Enhance proficiency in programming and software development, bolstering internet literacy and problem-solving skills.
	• Develop skills in various programming languages, understand software development life cycles, and learn effective debugging techniques.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply acquired knowledge of essential computer components, including the CPU, input/output devices, memory, and storage in understanding their roles in computer systems.
	• Analyze data using skills in internal data representation, including understanding number systems, binary arithmetic, Boolean algebra, and logic gates.
	• Evaluate computer organization, focusing on the roles of the CPU, memory, and secondary storage to gain a comprehensive understanding.
	• Design computer systems considering various input and output devices, their associated interfaces, and their functions within computer systems.
Content Outline	• **Knowing computer:** What is Computer, Basic Applications of Computer; Components of Computer System, Central Processing Unit (CPU), Input/output Devices, Computer Memory, Concepts of Hardware and Software; Concept of Computing, Data and Information, classification of computers, various generations of computers; What is an Operating System; Different Popular Operating Systems; The User Interface, System Software: System software Vs. Application Software, Types of System Software, Introduction and Types of Operating Systems • **Internal data Representation in Computers and Digital System Design:** Number Systems Used in Computers, Converting from One Number System to Another, Binary Arithmetic, Boolean Algebra, Boolean Functions, Logic Gates, Logic Circuits • **Computer Architecture:** Basic Functions of a Computer, Basic Computer Organization, CPU Architectures, Memory Architectures, Secondary Storage, Classification of Secondary Storage, Memory Storage Devices, • **Input- Output Device**s :Input Devices, Output Devices, I/O Interfaces
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the evolution and various types of

	computer languages, using them appropriately in different programming scenarios.
	• Analyze programming challenges using skills in programming constructs, algorithms, and flowcharting for effective problem-solving and program development.
	• Evaluate the roles of language processors in software development, understanding their impact on program execution and performance.
	• Design software solutions with a clear understanding of the relationship between software and hardware in the functioning of computer systems.
Content Outline	• **Computer Languages:** Definition, Generations of computer languages, Types of Languages, Language Processors: Assembler, Interpreter, Compiler, Linker and Loader. Programming constructs, Algorithm & flowchart. • **Software:** Basic Concepts and Terminologies, What is Software? ,Relationship between Hardware and Software, Software Development Life Cycle (SDLC), Advantages of SDLC Model, Limitations of SDLC Model, Software Testing and Debugging • **Introduction to Internet:** WWW and Web Browsers: Basic of Computer networks; LAN, WAN; Concept of Internet; Applications of Internet; connecting to internet; What is ISP; Knowing the Internet; Basics of internet connectivity related troubleshooting, World Wide Web; Web Browsing software, Search Engines; Understanding URL; Domain name; IP Address; Using e-governance website

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:
Activity 1: Write an essay on the essential components of a computer system, including the CPU, input/output devices, memory, and storage, explaining their roles and functions.
Activity 2: Create a presentation on internal data representation, explaining number systems, binary arithmetic, Boolean algebra, and logic gates. Include examples and illustrations to enhance understanding.
Activity 3: Design a basic computer system on paper, considering various input and output devices and their associated interfaces. Explain the function of each component within the system.

Module 2:
Activity 1: Write a report on the evolution of computer languages. Discuss different types of languages and their appropriate uses in different programming scenarios. Activity 2: Choose a programming problem and develop a solution using programming constructs, algorithms, and flowcharting. Explain your process and how you arrived at your solution.
Activity 3: Create a presentation on the role of language processors in software development. Discuss their impact on program execution and performance.

References:

1. Sinha, P. K., & Sinha, P. (2004). Computer Fundamentals. BPB Publications.

2. Srivastava, C. (2000). Fundamentals of Information Technology. Kalyani Publishers.
3. Rajaraman, V. (2003). Fundamentals of Computers (4th ed.). PHI Publication.
4. Jain, R. K. (2014). Fundamentals of Programming. S. Chand Publication.

1.5 Major (Elective-I) -A

Course Title	**Cyber Security**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of various cyber threats, including Cyber Warfare, Cyber Crime, Cyber Terrorism, and Cyber Espionage, differentiating their unique characteristics and potential impacts.
	• Analyze the importance of a comprehensive Cyber Security policy, the role of a nodal authority, and the significance of an international convention on Cyberspace for global cybersecurity standards.
	• Evaluate vulnerabilities in software, system administration, network architectures, data access, authentication, broadband communications, and human factors, devising strategies to mitigate these risks.
	• Design basic security measures for HTTP and SOAP services, understanding identity management and authorization patterns, and addressing challenges in securing web applications.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply knowledge of Cyber Security to identify and understand vulnerabilities in software, system administration, network architectures, and other areas, devising strategies to mitigate these risks.
	• Analyze systems and networks by conducting security audits, identifying potential weaknesses, and recommending improvements.
	• Evaluate the role of cryptography in Cyber Security, applying cryptographic techniques to secure data communication and ensure data integrity and confidentiality.
	• Design proactive security measures by understanding the concept of ethical hacking, its role in identifying and addressing system and network vulnerabilities.
Content Outline	• **Introduction to Cyber Security** Overview of Cyber Security, Internet Governance–Challenges and Constraints, Cyber Threats:- Cyber Warfare-Cyber Crime-Cyber, terrorism-Cyber Espionage, Need for a Comprehensive Cyber Security Policy, Need for a Nodal Authority, Need for an International convention on Cyberspace • **Cyber Security Vulnerabilities and Cyber Security Safeguards** Cyber Security Vulnerabilities Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards-Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**

	• Apply fundamental security measures for HTTP applications, protecting against common vulnerabilities and ensuring data integrity and safety.
	• Analyze identity management principles and implement secure identity practices within web services, focusing on user verification and access control.
	• Evaluate the effectiveness of applied security measures in preventing and detecting unauthorized access attempts by external entities.
	• Design robust security protocols to safeguard network and system resources, enhancing overall cybersecurity posture.
Content Outline	• **Securing Web Application** Services and Servers Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges. • **Intrusion Detection and Prevention** Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, System Integrity Validation
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply knowledge of both symmetric and asymmetric key cryptography, understanding their unique applications in securing data transmission and storage.
	• Analyze data security requirements in various contexts and utilize appropriate cryptographic methods to secure the data.
	• Evaluate the effectiveness of cryptographic methods in different applications, ensuring data security in various contexts.
	• Design strategies for data security by understanding the specificities of the Indian cyber space, including its regulatory framework, challenges, and initiatives.
Content Outline	• **Cryptography and Network Security:** Introduction to Cryptography, Symmetric key Cryptography, Asymmetric key Cryptography, Message Authentication, Digital Signatures, Applications of Cryptography. Overview of Firewalls- Types of Firewalls, User Management, VPN Security, Security Protocols:- security at the Application Layer-PGP and S/MIME, Security at Transport Layer-SSL And TLS, Security at Network Layer-IPSec. • **Cyber space and the Law**: Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyber space, Cyber Security Standards. The INDIAN Cyber space, National Cyber Security Policy 2013.
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply skills in initiating and conducting preliminary investigations in response to suspected cyber incidents, ensuring the preservation of digital evidence.
	• Analyze digital evidence stored on computer hard drives and other storage media by developing proficiency in conducting disk-based analysis.

	• Evaluate the results from the preliminary investigations and disk-based analysis, ensuring accuracy and completeness for potential forensic analysis.
	• Design effective strategies for preserving and analyzing digital evidence, crucial for investigating and responding to cybersecurity incidents.
Content Outline	• **Cyber Forensics:** Introduction to Cyber Forensics, Handling Preliminary Investigations, Controlling an Investigation, Conducting disk-based analysis, Investigating Information-hiding, Scrutinizing E-mail, Validating E mail header information, Tracing Internet access, Tracing memory in real-time.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write a report on a recent cyber security incident in which you identify vulnerabilities that were exploited and devise potential strategies to mitigate these risks.

Activity 2: Conduct a security audit of a hypothetical system or network, identifying potential weaknesses and recommending improvements. Discuss the role of cryptography in securing data communication and data integrity within this system or network.

Module 2:

Activity 1: Design a set of fundamental security measures for a hypothetical HTTP application. Discuss how these measures protect against common vulnerabilities and ensure data integrity and safety.

Activity 2: Write a report on identity management principles and their application within web services. Include a discussion on user verification and access control, and how they can prevent unauthorized access attempts.

Module 3:

Activity 1: Write a short essay on symmetric and asymmetric key cryptography. Discuss their unique applications in securing data transmission and storage.

Activity 2: Choose a data security requirement for a hypothetical scenario and discuss which cryptographic method would be most appropriate to secure the data. Evaluate the effectiveness of this method in the given context.

Module 4:

Activity 1: Write a mock incident report for a suspected cyber incident. Discuss how you would initiate and conduct a preliminary investigation to preserve digital evidence.

Activity 2: Analyze a hypothetical digital evidence stored on a computer hard drive. Discuss your findings and their potential implications for a forensic analysis.

References:

1. Hassan, N., & Hijazi, R. (n.d.). Digital Privacy and Security Using Windows: A Practical Guide. Apress.
2. DSCI-Nasscom. (2013). Cyber Crime Investigation.
3. Gobole, N. (2008). Information Systems Security: Security Management, Metrics, Frameworks and Best Practices (With CD).
4. Weber, R. (1999). Information Systems Control and Audit. Pearson Pub.
5. Pettier, T. (2004). Information Security Policies, Procedures and Standards.
6. Tipton, H. F. (Ed.). (2003). Information Security Management Handbook (5th Edition).
7. Basta, A., & Halton, W. (2008). Computer Security. Cengage Learning
8. Peltier, T. R. (2016). Information Security Policies. Auerbach Publications

1.5 Major (Elective-I) -B

Course Title	**Computer Oriented Statistical Techniques**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply knowledge of statistical analysis by calculating and interpreting averages, including arithmetic mean, weighted arithmetic mean, median, and mode for both raw and grouped data.
	• Analyze data using measures of central tendency to gain insights into the data's overall pattern, which are fundamental in statistical analysis and interpretation.
	• Understand and evaluate summation notation and other essential mathematical notations used in statistical analysis for accuracy in conducting complex statistical calculations.
	• Design data interpretation strategies based on the understanding of mathematical expressions in statistical analysis, ensuring accurate and meaningful data analysis results.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of measures of central tendency and dispersion, calculating them using raw and grouped data, and making necessary adjustments like Sheppard's correction for variance.
	• Analyze and compare different measures of central tendency and dispersion, interpreting their relationships and significance in various contexts.
	• Evaluate the accuracy and effectiveness of these statistical measures and software tools used in data analysis.
	• Design comprehensive statistical reports to summarize and interpret data, aiding in data-driven decision making.

Content Outline	• **The Mean, Median, Mode, and Other Measures of Central Tendency:** Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean , The Weighted Arithmetic Mean ,Properties of the Arithmetic Mean, The Arithmetic Mean Computed from Grouped Data ,The Median ,The Mode, The Empirical Relation Between the Mean, Median, and Mode, The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency. • **The Standard Deviation and Other Measures of Dispersion:** Dispersion, or Variation, The Range, The Mean Deviation, The Semi- Interquartile Range, The Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation. • Charlie's Check, Sheppard's Correction for Variance, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures of Dispersion.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of moments, skewness, kurtosis, probability definitions, and sampling theory in statistical analysis.
	• Analyze data by computing moments, skewness, and kurtosis, performing probability calculations, and analyzing sampling distributions using software tools.
	• Evaluate relationships between statistical measures and create visual and computational representations of data to understand data trends and patterns.
	• Design strategies for data analysis by leveraging understanding of statistical measures and software tools.
Content Outline	• **Moments, Skewness, and Kurtosis:** Moments , Moments for Grouped Data ,Relations Between Moments , Computation of Moments for Grouped Data, Charlie's Check and Sheppard's Corrections, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness, and Kurtosis, Software Computation of Skewness and Kurtosis. • **Elementary Probability Theory:** Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Probability Distributions, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinatorial Analysis, Combinations, Stirling's Approximation to n!, Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability. • **Elementary Sampling Theory:** Sampling Theory, Random Samples and Random Numbers, Sampling With and Without Replacement, Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences and Sums, Standard Errors, Software Demonstration of Elementary Sampling Theory.
Module 3 (Credit 1)	

Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of key concepts in statistical estimation theory and decision theory, including estimation parameters, hypothesis testing, significance levels, and distributions.
	• Analyze data using estimation methods, hypothesis tests, and R programming to calculate and visualize statistical measures and distributions, assessing their reliability and significance in various scenarios.
	• Evaluate the performance and appropriateness of statistical methods, designing statistical studies and developing R scripts to automate and visualize data analysis.
	• Design statistical analysis strategies using R scripting and other statistical methods to understand and interpret data trends and patterns.
Content Outline	• **Statistical Estimation Theory:** Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters and Performance, Probable Error. • **Statistical Decision Theory:** Statistical Decisions, Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II Errors, Level of Significance, Tests Involving Normal Distributions, Two-Tailed and One-Tailed Tests, Special Tests, Operating-Characteristic Curves; the Power of a Test, p-Values for Hypotheses Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions. • **Statistics in R**:mean, median, mode, Normal Distribution, Binomial Distribution, Frequency Distribution in R
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply knowledge of the freehand method for curve fitting, discussing its applications and limitations.
	• Analyze data using the derivation of the least-squares line for curve fitting.
	• Evaluate the concept of the straight line in curve fitting, understanding its role and implications.
	• Design strategies for data analysis using the method of least squares for linear regression in curve fitting.
Content Outline	• **Curve Fitting and the Method of Least Squares:** Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least-Squares Line, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables • **Correlation Theory:** Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation, Remarks Concerning the Correlation Coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear

	Correlation Coefficient, Correlation of Time Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Calculate the mean, median, mode, and other measures of central tendency and dispersion for a given data set. Discuss the relationships and significance of these measures in the context of the data set.

Activity 2: Use a statistical software tool to calculate these measures for a larger, more complex data set. Write a comprehensive statistical report summarizing and interpreting the data.

Module 2:

Activity 1: Calculate the moments, skewness, and kurtosis for a given data set. Discuss the relationships between these measures and their significance in the context of the data set.

Activity 2: Write a program in R to perform probability calculations and analyze a sampling distribution. Discuss the results and their implications.

Module 3:

Activity 1: Choose a parameter to estimate for a given data set. Calculate the point estimate and interval estimate for this parameter and discuss their reliability.

Activity 2: Formulate a hypothesis for a given scenario. Use R to test this hypothesis, discussing the results and their significance.

Module 4:

Activity 1: Use the freehand method to fit a curve to a given data set. Discuss the applicability and limitations of this method.

Activity 2: Use the method of least squares to fit a linear regression line to a given data set. Discuss the results and their implications.

References:

1. Newbold, P., Carlson, W. L., & Thorne, B. (2013). Statistics for Business and Economics. Pearson Publication.
2. Johnson, R. A., & Wichern, D. W. (2012). Applied Multivariate Statistical Analysis. Prentice Hall India Learning Private Limited.
3. Ruppert, D. (2015). Statistics and Data Analysis for Financial Engineering. Springer.
4. Agresti, A., & Finlay, B. (2009). Statistical Methods for the Social Sciences. Pearson.

5. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). An Introduction to Statistical Learning. Springer Texts in Statistics.
6. Wickham, H., & Grolemund, G. (2017). R for Data Science. Greyscale Indian Edition.

1.6 Minor Stream (RM)

Course Title	**Research Methodology**
Course Credits	**4 Credits**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of the key steps involved in the research process, including problem formulation, literature review, data collection, analysis, and interpretation.
	• Analyze research challenges by formulating clear and focused research questions and hypotheses based on a thorough review of existing literature and identification of research gaps.
	• Evaluate and select suitable research designs based on the nature of the research questions, including experimental, quasi-experimental, and non-experimental designs.
	• Design comprehensive literature reviews to identify relevant studies, theories, and methodologies within a specific research domain.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the fundamental meaning of research and its role in acquiring knowledge and solving problems, identifying the primary objectives of research, including the pursuit of knowledge, problem solving, and contributing to existing knowledge.
	• Analyze research challenges by developing the skills necessary to define a clear and focused research problem, including the identification of gaps in existing literature and formulating relevant research questions.
	• Evaluate hypotheses as they serve as a tentative explanation or prediction that guides the research, understanding their role in the research process.
	• Design research strategies based on a clear understanding of research fundamentals, problem identification, and hypothesis formulation.
Content Outline	• **Foundations of Research:** Meaning, Objectives, Motivation, Utility. Concept of theory, empiricism, deductive and inductive theory. Characteristics of scientific method – Understanding the language of research – Concept, Construct, Definition, Variable Research Process. • **Problem Identification & Formulation** – Research Question – Investigation Question. • **Measurement Issues** – Hypothesis – Qualities of a good Hypothesis –Null Hypothesis & Alternative Hypothesis. Hypothesis Testing – Logic & Importance
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of research design and its crucial role in shaping the structure and execution of research studies.
	• Analyze research challenges using the concept of exploratory research, its types, and its applications in uncovering new insights and generating hypotheses.

	• Evaluate the use of qualitative research, with its emphasis on exploring the depth and complexity of phenomena. Understand the concepts of measurement, causality, generalization, and replication in the context of qualitative research.
	• Design research strategies considering the concept of scaling in research, recognizing its importance in measuring attitudes, opinions, and other abstract concepts.
Content Outline	• **Research Design:** Concept and Importance in Research – Features of a good research design – Exploratory Research Design – concept, types and uses, Descriptive Research Designs – concept, types and uses. Experimental Design: Concept of Independent & Dependent variables. • **Qualitative and Quantitative Research:** Qualitative research – Quantitative research – Concept of measurement, causality, generalization, replication. Merging the two approaches. • **Measurement and Scale:** Concept of measurement– what is measured? Problems in measurement in research – Validity and Reliability. Measurement Scale – Nominal, Ordinal, Interval, Ratio.
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the sampling frame and its role as a list or framework from which the actual sample is drawn.
	• Analyze the factors influencing sample size determination and its significance in achieving reliable results.
	• Evaluate the key characteristics that contribute to the representativeness, reliability, and validity of a good sample in research.
	• Design research strategies considering the process of selecting a simple random sample, understanding its application in ensuring each element has an equal chance of being included.
Content Outline	• **Sampling:** Concepts of Statistical Population, Sample, Sampling Frame, Sampling Error, Sample Size, Non Response. Characteristics of a good sample. • **Probability Sample** – Simple Random Sample, Systematic Sample, Stratified Random Sample & Multi-stage sampling. Determining size of the sample, Practical considerations in sampling and sample size.
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the importance of data preparation in ensuring the quality and reliability of data for analysis.
	• Analyze the essential components and structure of a research paper, including title, abstract, introduction, methodology, results, discussion, and conclusion.
	• Evaluate the effective use of reference materials such as encyclopedias, research guides, handbooks, and other resources to enhance research quality.
	• Design data science findings presentations using effective communication skills, leveraging knowledge gained through case study analysis.

Content Outline	• **Analysis:** Data Preparation – Univariate analysis (frequency tables, bar charts, pie charts, percentages), Bivariate analysis – Cross tabulations and Chi-square test including testing hypothesis of association. • **Interpretation of Data and Paper Writing** – Layout of a Research Paper, Journals in Computer Science, Impact factor of Journals, When and where to publish ? Ethical issues related to publishing, Plagiarism and Self Plagiarism. • **Use of Encyclopedias**, Research Guides, Handbook etc., Academic Databases for Computer Science • **Business Communication** for Data Science (Case Study-Presentation)

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write a short essay on the fundamental meaning of research, its role in acquiring knowledge and solving problems. Discuss the primary objectives of research.

Activity 2: Identify a research problem in your area of interest. Define the problem clearly, identify gaps in existing literature, and formulate relevant research questions and hypotheses.

Module 2:

Activity 1: Design a simple research study. Choose whether to use an exploratory, descriptive, or experimental design and explain your choice. Discuss how you would approach qualitative or quantitative research in your study.

Activity 2: Create a measurement scale for a variable in your research study. Discuss the type of scale you chose (nominal, ordinal, interval, ratio) and why.

Module 3:

Activity 1: Explain the concept of a sampling frame and its importance in research. Discuss the factors that influence sample size determination.

Activity 2: Select a sampling method for your research study and explain why you chose it. Discuss how you would ensure your sample is representative.

Module 4:

Activity 1: Discuss the importance of data preparation in research. Describe the steps you would take to ensure your data is reliable for analysis.

Activity 2: Write a mock research paper based on your study design from the previous modules. Be sure to include all essential components: title, abstract, introduction, methodology, results, discussion, and conclusion.

Activity 3: Create a presentation of your research findings. Practice presenting your findings in a clear and concise manner.

References

1. Kothari, C. R. (2004). Research Methodology. Wiley Eastern.
2. Wilkinson, K. P., & Bhandarkar, L. (2010). Formulation of Hypothesis. Himalaya Publication.
3. Best, J. W., & Kahn, V. (2016). Research in Education. PHI Publication.
4. Kumar, R. (2023). Research Methodology: A Step by Step Guide for Beginners. Pearson.
5. Krishna Swami, K. N., & Others. (2006). Management Research Methodology: Integration of Principles, Methods, and Techniques. Pearson Education.

Semester II

2.1 Major (Core)

Course Title	**Data Structures and Analysis of Algorithm**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of algorithm analysis, including asymptotic performance, to assess and optimize algorithm efficiency.
	• Analyze algorithms by writing rigorous correctness proofs, ensuring their accuracy and validity.
	• Demonstrate familiarity with major algorithms and data structures, understanding their application in various scenarios.
	• Design efficient algorithms using important algorithmic design paradigms and methods of analysis, synthesizing these in software design and development.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the fundamental concepts of data structures, including arrays, stacks, queues, and linked lists.
	• Analyze scenarios where each data structure is best suited for efficient data organization and manipulation, choosing the appropriate data structure based on the situation.
	• Evaluate algorithmic time and space complexities using a clear understanding of asymptotic notations, including Big-Oh, Omega, and Theta.
	• Design efficient solutions to data-related problems by applying asymptotic notations to evaluate and compare the efficiency of algorithms related to data structures.
Content Outline	• **Introduction to Data Structures** – Fundamental Elements – Asymptotic Notations: Big-Oh, Omega and Theta – Best, Worst and Average case Analysis: • **Definition and an example** Arrays and its representations • **Stacks and Queues** – Linked lists – Singly Linked List – Doubly linked list – Linked list based implementation of Stacks and Queues. Evaluation of Expressions
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the basic concepts of trees and their significance in data representation and manipulation.
	• Analyze differences between trees and other data structures, emphasizing the hierarchical nature of tree structures.
	• Evaluate the use of binary trees and their applications in various domains.
	• Design binary trees using both array and linked list structures, demonstrating proficiency in these representations.
Content Outline	• **Trees:** Introduction to Trees – Basic concepts – Binary Trees – Binary tree representations (Array and list) and Traversals Techniques (Preorder, Inorder, Postorder) • **Succinct Data Structures**: Overview – Level order representation of Binary Trees – Rank and Select – Sub trees. • **Graphs:** Definitions, Terminologies, Matrix and Adjacency List

	Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. • **Searching and sorting:** Sorting Algorithms: Basic concepts – Bubble Sort – Insertion Sort – Selection Sort – Quick Sort – Shell sort – Heap Sort – Merge Sort – External Sorting. Searching: Linear Search, Binary Search. • **Indexing:** Hashing – Hash Functions – Separate Chaining – Open Addressing: Linear Probing- Quadratic Probing- Double Hashing- Rehashing – Extendible Hashing.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

1. Test student's understanding of fundamental concepts of data structure and algorithm.
2. Research and present examples illustrating Big-Oh, Omega, and Theta notations.
3. Analyze and compare the time complexities of different algorithms using these notations.

Module 2:

1. Create and visualize binary trees using array and linked list representations.
2. Implement and demonstrate tree traversal techniques (Preorder, Inorder, Postorder).
3. Explore and implement Breadth First Search and Depth First Search algorithms for graph traversal.
4. Study and implement level order representation of binary trees.
5. Investigate and implement Rank and Select operations on succinct data structures.
6. Implement Binary Search Trees and AVL Trees.
7. Analyze and compare the performance of these tree structures.
8. Implement hash functions and explore different hashing techniques.

References

1. Weiss, M. A. (2003). Data Structures and Algorithm Analysis in C (2nd ed.). Pearson Education.
2. Pai, G. A. V. (2008). Data Structures and Algorithms: Concepts, Techniques, and Applications (1st ed.).
3. Horowitz, E., Sahni, S., & Anderson-Freed, S. (2007). Fundamentals of Data Structures in C (2nd ed.). University Press.
4. Tremblay, J.-P., & Sorenson, P. G. (2007). An Introduction to Data Structures with Applications (2nd ed.). Tata McGraw-Hill.
5. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2003). Introduction to Algorithms (2nd ed.). PHI.
6. Gilberg, R., & Forouzan, B. (2007). Data Structures: A Pseudo-code Approach with C. Thomson Learning.

7. Dave, P., & Dave, H. (2008). Design and Analysis of Algorithms. Pearson Education.
8. Berman, M. (2004). Data Structures Via C++: Objects by Evolution. Oxford Univ. Press.
9. Knuth, D. E. (1973). Sorting and Searching: The Art of Computer Programming, Vol. 3. Addison-Wesley Publishing.
10. Kanetkar, Y. (2003). Data Structures Through C. BPB publications.

2.1 Major Core

Course Title	**Data Structures and Analysis of Algorithm (Lab)**
Course Credits	**2**
Course Outcomes	**After completion of this Course, the students will be able to**
	• Apply important algorithmic design paradigms, like divide and conquer, dynamic programming, and greedy methods, to solve complex problems.
	• Analyze algorithms using methods such as worst-case, average-case, and best-case analysis, amortized analysis, and complexity analysis (using Big O, Omega, and Theta notations), to ensure optimal performance and efficiency.
Module1 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Implement singly linked lists, which consist of nodes where each node contains a data field and a reference(link) to the next node in the sequence.
	• Apply understanding of doubly linked lists, where each node contains a reference to the next node as well as the previous node, offering more flexibility in navigating the list but increasing complexity in modification operations.
Content Outline	• Implement Stacks and Queues (Linked lists, Singly Linked List, Doubly linked list)
Module2 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply understanding of tree traversal techniques including Preorder (visit the root first, then the left subtree, and finally the right subtree), Inorder (visit the left subtree first, then the root, and finally the right subtree), and Postorder (visit the left subtree first, then the right subtree, and finally the root).
	• Implement hash functions, which are used to map data of arbitrary size to fixed-size values. Hash functions are key in many data structures and algorithms, including hash tables, digital signatures, and data retrieval.

Content Outline	• Implement and demonstrate tree traversal techniques (Preorder, Inorder, Postorder). • Explore and implement Breadth First Search and Depth First Search algorithms for graph traversal. • Analyze and compare the performance of these tree structures. • Implement hash functions and explore different hashing techniques.

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write a program to implement a singly linked list. The program should be able to add elements, delete elements, and display the linked list.

Activity 2: Write a program to implement a doubly linked list. Like the previous program, it should be able to add elements, delete elements, and display the linked list. Compare the two programs and discuss the advantages and disadvantages of using singly linked lists versus doubly linked lists.

Module 2:

Activity 1: Write a program to implement a binary tree. The program should be able to add elements and perform tree traversal techniques including Preorder, Inorder, and Postorder.

Activity 2: Write a program to implement hash functions. The program should be able to map data of arbitrary size to fixed-size values. Discuss different hashing techniques and their uses.

Activity 3: Implement Breadth First Search and Depth First Search algorithms for graph traversal. Analyze and compare the performance of these traversal techniques.

References

11. Weiss, M. A. (2003). Data Structures and Algorithm Analysis in C (2nd ed.). Pearson Education.
12. Pai, G. A. V. (2008). Data Structures and Algorithms: Concepts, Techniques, and Applications (1st ed.).
13. Horowitz, E., Sahni, S., & Anderson-Freed, S. (2007). Fundamentals of Data Structures in C (2nd ed.). University Press.
14. Tremblay, J.-P., & Sorenson, P. G. (2007). An Introduction to Data Structures with Applications (2nd ed.). Tata McGraw-Hill.
15. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2003). Introduction to Algorithms (2nd ed.). PHI.
16. Gilberg, R., & Forouzan, B. (2007). Data Structures: A Pseudo-code Approach with C. Thomson Learning.

17. Dave, P., & Dave, H. (2008). Design and Analysis of Algorithms. Pearson Education.
18. Berman, M. (2004). Data Structures Via C++: Objects by Evolution. Oxford Univ. Press.
19. Knuth, D. E. (1973). Sorting and Searching: The Art of Computer Programming, Vol. 3. Addison-Wesley Publishing.
20. Kanetkar, Y. (2003). Data Structures Through C. BPB publications.

2.2 Major (Core)

Course Title	**Data Communications and Networking**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of key concepts in data communications, including digital and analog signals, bandwidth, data transmission modes, and network types.
	• Analyze the OSI (Open Systems Interconnection) model and TCP/IP protocol suite, understanding their structure, functions, and importance in data communication and networking.
	• Evaluate different networking protocols such as TCP, UDP, IP, and HTTP, understanding their functions, characteristics, and application scenarios.
	• Design and implement basic computer networks based on specific requirements, including choosing the appropriate networking equipment and setting up connections.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the concept of computer networks and their significance in modern computing, including data sharing, resource sharing, and communication efficiency.
	• Analyze the purposes and advantages of connecting computers in a networked environment such as improved collaboration, resource sharing, and cost-effectiveness.
	• Differentiate between common network topologies like bus, star, ring, mesh, and tree, understanding their structure and working principles.
	• Evaluate the strengths and weaknesses of various topologies in different scenarios, considering factors like scalability, fault tolerance, installation and maintenance costs.
Content Outline	• **Introduction:** Computer Networks and its uses, Network categorization and Hardware: Broadcast and point-to-point networks, Local Area Network (LAN), Metropolitan Area Network (MAN), Wide Area Networks (WAN), Inter networks, Topologies, Wireless Networks, Network Software: Protocols, Services, network architecture, design issues, OSI Reference model, TCP/IP Reference model, Comparison of OSI and TCP/IP Models. Introduction to Example Networks: Internet, Connection Oriented Networks–X.25, Frame Relay, ATM • **Data Communication Model**: Digital and Analog data and signals, bit rate, baud, bandwidth, Nyquist bit rate, Guided Transmission Media – Twisted Pair, Coaxial cable, Optical fiber; wireless transmission–Radio waves, microwaves, infrared waves; Satellite Communication.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Differentiate circuit switching from other switching techniques like packet switching and message switching. Circuit switching involves establishing a dedicated communication path between devices for the duration of their conversation.
	• Describe the process of establishing a circuit in a circuit-switched network, which includes call setup, data transfer, and call teardown phases.

	• Analyze the advantages and disadvantages of circuit switching. Advantages include predictable performance and minimal delay, while disadvantages include inefficiency in resource utilization and potential for wasted capacity.
	• Understand common circuit switching protocols such as Integrated Services Digital Network (ISDN), which is a set of communication standards for simultaneous digital transmission of voice, video, data, and other network services.
Content Outline	• **Switching:** Circuit Switching, Packet switching; Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Synchronous and Asynchronous TDM, Modems, Transmission impairments, Manchester and differential Manchester encoding • **Error Detection and Correction:** Types of errors Redundancy, Detection Versus Correction, Error Detection, Error Correction, Hamming Code, Cyclic Redundancy Check, Check sum and Its idea.
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the role and functions of the Data Link Layer in the OSI model, as well as its relationship with the Physical Layer.
	• Analyze various framing techniques used in the Data Link Layer, including character count, flag bytes with byte stuffing, and bit stuffing.
	• Evaluate the efficiency and effectiveness of these framing techniques in ensuring accurate and reliable data transmission.
	• Design and implement framing algorithms for efficient data encapsulation and transmission across networks.
Content Outline	• **Data Link Layer Design issues:** Framing, error control, Flow Control, Error Detection and correction; Elementary Data Link Protocols, Sliding Windows Protocols; Medium Access Control: Aloha, CSMA protocols, Collision free protocols, Limited Contention Protocols; Wave length division Multiple access protocol, Wireless • **LAN Protocol:** MACA; IEEE 802.3Ethernet, IEEE 802.4 Token Bus; IEEE 802.5 Token ring, Binary Exponential Back off algorithm • **Digital Cellular, Radio :** Global System for Mobile, Communication (GSM), Code Division Multiple Access (CDMA)
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the responsibilities of the Network Layer in the OSI model and differentiate it from other layers.
	• Analyze and design efficient addressing schemes for packet routing, implementing them as per network requirements.
	• Evaluate the performance, advantages, and disadvantages of virtual circuit switching compared to other switching techniques.
	• Design network configurations using appropriate switching techniques to meet specific networking needs.

Content Outline	• Network Layer, Design issues Virtual circuit and Datagram Subnet, Routing Algorithms, Optimality principle, shortest path routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcast and Multi Cast Routing, Routing for Mobile hosts, Routing in Ad hoc Networks • Congestion Control Algorithm, General Principals Traffic Shaping, Leaky Bucket, Token Bucket, choke packets, Load Shedding

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Test students' understanding of fundamental concepts in data communications and networking.

Module 1:

Activity 1: Write a report discussing the concept of computer networks and their significance in modern computing. Discuss the advantages of connecting computers in a networked environment and differentiate between common network topologies like bus, star, ring, mesh, and tree.

Activity 2: Choose a specific network topology and evaluate its strengths and weaknesses in a given scenario. Consider factors like scalability, fault tolerance, installation, and maintenance costs.

Module 2:

Activity 1: Write a report differentiating between circuit switching and other switching techniques like packet switching and message switching. Discuss the process of establishing a circuit in a circuit-switched network and analyze the advantages and disadvantages of circuit switching.

Activity 2: Design a simple circuit-switched network using a common circuit switching protocol. Document and reflect on your process.

Module 3:

Activity 1: Write a report on the role and functions of the Data Link Layer in the OSI model. Analyze various framing techniques used in the Data Link Layer and evaluate their efficiency and effectiveness.

Activity 2: Implement a simple framing algorithm for efficient data encapsulation and transmission across networks. Document and reflect on your process and any challenges you encountered.

Module 4:

Activity 1: Write a report discussing the responsibilities of the Network Layer in the OSI model and differentiating it from other layers. Discuss efficient addressing schemes for packet routing.

Activity 2: Design a network configuration using an appropriate switching technique to meet a specific networking need. Reflect on the performance, advantages, and disadvantages of your chosen technique

References: -

1. Forouzan, B. A. (2007). Data Communications and Networking (4th ed.). McGraw Hill.
2. Stallings, W. (2013). Data and Computer Communications (10th ed.). Pearson.
3. Tanenbaum, A. S. (2010). Computer Networks (5th ed.). Pearson.

2.3. Major Core

Course Title	**Web Technology**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of HTML forms and various input elements to design and implement user-interactive web pages, including form validation processes.
	• Analyze the need for responsive web design, developing web pages using HTML and CSS that ensure optimal viewing experience across different devices and screen sizes.
	• Evaluate the configuration and administration of a web server, discussing the concept of virtual hosting and its implications.
	• Design and implement a web server configuration, including the setup of virtual hosting as per specific requirements.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of HTTP basics, client requests, server responses, and common HTTP headers, discussing the significance of persistent connections and cookies in session management.
	• Analyze and create well-formed and valid XML documents, defining Document Type Definitions (DTD) and exploring the Document Object Model (DOM) in XML.
	• Evaluate the need for dynamic and interactive web pages in modern web development.
	• Design and develop dynamic and interactive web pages using JavaScript, including the use of variables, functions, loops, and conditional statements.
Content Outline	• **Introduction to Web Technologies**: Concepts of Internet, Concepts of World Wide Web, Internet based Services Email, Telnet, FTP,WWW,Web Server, Web Hosting, DNS, SMTP. • **HTML:** Introduction to HTML, Structure of HTML document, Basic HTML tags, Meta tags Introduction to HTML5. • **Angular JS:** Environment Setup, Creating and executing angular js application, directives, • controllers, expressions, filters, tables, modules, forms, views, scopes, services. • CSS: Introduction to CSS, Types of CSS- use of & in CSS, Introduction of CSS3 : Gradients, Transitions, Animations, multiple columns. • **XML:** Introduction to XML, Valid and Well- Defined Document, Document Type Definition or DTD, uses of DTD, XML Tags, Elements, Attributes, PCDATA, CDATA, XML Schema, Defining, Accessing XML Document. • **Clint Side Scripting Language: Javascript**: Introduction to javascript, Variables, identifiers constants, Operators in javascripts, Event handling javascript, Validations in JavaScript.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the role of server-side scripting in web development and differentiate various server-side scripting languages and their use cases.
	• Analyze the need for server-side logic, database interactions, and dynamic content generation in web development.

	• Evaluate the strengths and weaknesses of PHP as a server-side scripting language.
	• Design and implement server-side scripts using PHP to handle server-side logic, database interactions, and dynamic content generation.
Content Outline	• **JQuery:** Introduction to JQuery, Selectors, attributes, Traversing, CSS, DOM, Events, AJAX, Effects, Interactions, Widgets,Theming. • **Server Side Scripting Language: PHP** :Configuration and Installation, Basic Structure of PHP script COOKIE variables, Sending E mail, Database Operations with PHP, Connecting to My-SQL. • **CMS:** Wordpress • Introduction to CGI Programming, JSP, Servlet, AJAX. Creation of .jar project. Deployment of Java application on Appserver.

Assignments/Activities to wards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Write a report discussing HTTP basics, client requests, server responses, and common HTTP headers. Discuss the significance of persistent connections and cookies in session management. Include examples where possible.

Activity 2: Design and develop a dynamic and interactive web page using HTML5 and JavaScript. The page should include the use of variables, functions, loops, and conditional statements. The HTML should be well-formed and valid.

Module 2:

Activity 1: Write a report comparing various server-side scripting languages and their use cases. Discuss the role of server-side scripting in web development, the need for server-side logic, database interactions, and dynamic content generation. Evaluate the strengths and weaknesses of PHP as a server-side scripting language.

Activity 2: Implement a simple server-side script using PHP. The script should handle server-side logic, interact with a database, and generate dynamic content. Document and reflect on your process and any challenges you encountered.

References

1. Lecky Thompson, G. W. (2009). Web Programming. Cengage Learning.
2. Powell, T. (2003). Web Design: The Complete Reference. Tata McGraw Hill.
3. Powell, T. (2008). HTML and XHTML: The Complete Reference. Tata McGraw Hill.
4. Powell, T., & Schneider, F. (2004). JavaScript2.0: The Complete Reference (2nd ed.).
5. Holzner, S. (2007). PHP: The Complete Reference. Tata McGraw Hill.
6. Duckett, J. (2004). Beginning Web Programming with HTML, XHTML, CSS & JavaScript. Wrox.
7. Spainhour, S. (2002). Webmaster in a Nutshell. O'Reilly and Associates.
8. Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly and Associates.
9. Buser, D., et al. (2000). Beginning ASP 3.0. Wrox.

2.3 Major (Core)

Course Title	**Web Technology (Lab)**
Course Credits	**2**
Course Outcomes	**After completion of this Course, the students will be able to**
	• Apply understanding of web technology concepts, including frontend, backend (like PHP), and database management.
	• Analyze and illustrate the PHP language, focusing on its syntax, structures, and use in server-side scripting.
	• Evaluate different web features, such as interactivity, responsiveness, and user interface design, illustrating their importance in web development.
	• Design and understand web services using MVC (Model-View-Controller) architecture, which separates an application into three interconnected components.
Module1 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply understanding of web development basics, exploring the concepts of HTML, CSS, and JavaScript for frontend development.
	• Analyze and illustrate the concept of MySQL for database management and PHP for server-side scripting.
	• Evaluate the strengths and weaknesses of PHP as a server-side scripting language, understanding its role in dynamic web development.
	• Design a local web development environment by installing and using the XAMPP web server.
Content Outline	• **Introduction** to Web Technology, HTML, Basic Tags, CSS, Table and Forms • Introduction to **JavaScript**, Variables, Operators, Data Type Conversions, functions, Control Structure, Date-Time functions and Form Manipulation • **MYSQL** – Introduction about Database, Data Types, DML, DDL, Aggregate functions, Data Time functions, Stored Procedure, Sub query and join • **PHP-**Introduction to PHP, History, Web Brower, Web Server, Xampp, Installation and Configuration files. Function, Arrays
Module 2 (Credit 1)	
Learning Outcomes	**After learning this module, learners will be able to**
	• Apply understanding of web features like sessions, forms, and cookies, which are crucial for user interaction and data handling on the web.
	• Analyze the concept of Ajax, a set of web development techniques for creating asynchronous web applications.

	• Evaluate Joomla as a content management system, demonstrating its use in web development.
	• Design web services using the MVC architecture in CakePHP, exploring its benefits in structuring and organizing code.
Content Outline	• **String and Patterns**-Quoting, Matching, Extracting, Searching, Replacing and Formatting • **Web Features**-Sessions, Forms, GET and POST data, Cookies, HTTP Headers. Database Programming. • **Streams and Network Programming**- Files, Reading, Writing, File System functions, Streams File Uploading and File Downloading. • **Ajax** Basics, Sending data to PHP with Ajax, Prototype- Utility functions, Ajax object and Form Object. Smarty variables, Variable Modifiers, Built-in Functions, custom functions, Config files. • Introduction to **Joomla** • **CakePHP** - MVC Overview, Naming Conversions, Model, V\view, Controller, Helpers, Scaffolding an Data Validation, Security, Web Services

Assignments/Activities to wards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity 1: Create a basic website using HTML, CSS, and JavaScript. The website should include various HTML elements styled with CSS and incorporate JavaScript to provide some sort of interactivity.

Activity 2: Using PHP, set up a form on your website that submits data to a MySQL database. Use XAMPP to run the server locally and test the functionality. Write a short evaluation of PHP as a server-side scripting language based on your experience.

Module 2:

Activity 1: Incorporate the use of sessions, forms, and cookies into your website from Module 1. For example, you could create a login form that starts a session and sets a cookie.

Activity 2: Create a simple web service using CakePHP and the MVC architecture. The service should use Ajax to communicate with the server asynchronously. Install Joomla and explore its features. Write a short report on your experience with Joomla as a content management system.

References: -

1. Lecky Thompson, G. W. (2009). Web Programming. Cengage Learning.
2. Powell, T. (2003). Web Design: The Complete Reference. Tata McGraw Hill.
3. Powell, T. (2008). HTML and XHTML: The Complete Reference. Tata McGraw Hill.

4. Powell, T., & Schneider, F. (2004). JavaScript2.0: The Complete Reference (2nd ed.).
5. Holzner, S. (2007). PHP: The Complete Reference. Tata McGraw Hill.
6. Duckett, J. (2004). Beginning Web Programming with HTML, XHTML, CSS & JavaScript. Wrox.
7. Spainhour, S. (2002). Webmaster in a Nutshell. O'Reilly and Associates.
8. Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly and Associates.
9. Buser, D., et al. (2000). Beginning ASP 3.0. Wrox.

2.4. Major Core

Course Title	**Introduction to AI**
Course Credits	**2**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of the role of AI in various fields such as healthcare, logistics, finance, and more, appreciating its transformative potential.
	• Analyze and practice programming in AI, using languages like Python or R and libraries like TensorFlow or PyTorch.
	• Evaluate the applicability and effectiveness of soft computing concepts, including fuzzy logic, neural networks, and genetic algorithms, in solving complex AI problems.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the role of AI in engineering and its applications in daily life, describing how it impacts various sectors and daily activities.
	• Analyze the difference between natural intelligence and artificial intelligence, discussing their characteristics and functionalities.
	• Evaluate various task domains of AI, identifying their significance and how they contribute to the overall functionality of AI systems.
	• Understand and discuss the limitations and challenges of AI, including ethical considerations, data privacy concerns, and technical hurdles.
Content Outline	• **Artificial Intelligence:** Role of AI in engineering, AI in daily life, Intelligence and Artificial Intelligence, Different task domains of AI, Programming methods, Limitations of AI Expert System • **Applications**: Phases in Building Expert System, Expert System Architecture, Expert System versus Traditional Systems, Rule based Expert Systems, Blackboard Systems, Truth Maintenance System, Application.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the concept of soft computing, appreciating its significance in solving complex problems where traditional methods may not be effective.
	• Analyze the distinctions between soft computing and hard computing approaches, explaining their differences, advantages, and disadvantages.
	• Evaluate the concepts of supervised and unsupervised learning, understanding their roles in machine learning and AI. Supervised learning involves learning from labeled data, while unsupervised learning involves finding patterns in unlabeled data.
Content Outline	• **Importance of soft computing** Soft computing versus hard computing; Supervised and unsupervised learning; • **Introduction to main components** of soft computing: Fuzzy logic, Neural networks, Genetic algorithms

Assignments/Activities towards Comprehensive Continuous Evaluation (CCE)

Module 1:
Activity: Write a report on the role of AI in a specific sector of engineering of your choice, and how it impacts daily activities in that sector. Analyze the difference between natural intelligence and artificial intelligence within the context of this sector. Discuss the significant task domains of AI in this sector and the limitations and challenges faced, including ethical considerations, data privacy concerns, and technical hurdles.

Module 2:
Activity: Choose a complex problem that traditional computing methods struggle to solve effectively. Apply your understanding of soft computing to propose a solution to this problem. Compare and contrast the soft computing approach with a hard computing approach to this problem. Analyze the advantages and disadvantages of each. Finally, design a simple supervised or unsupervised learning task that could be part of the solution, explaining why you chose that approach.

References:

1. Russel, S., & Norvig, P. (2015). Artificial Intelligence: A Modern Approach (3rd ed.). Pearson.
2. Khemani, D. (2013). A First Course in Artificial Intelligence (1st ed.). Tata McGraw-Hill.
3. Rich, E., Knight, K., & Nair, S. (2009). Artificial Intelligence (3rd ed.). Tata McGraw-Hill.
4. Deva, R. (2014). Artificial Intelligence: A Rational Approach (1st ed.). Shroff Publishers.
5. Das, A. B. (2013). Artificial Intelligence & Soft Computing for Beginners (1st ed.). SPD.

2.5 Major (Elective-II)-A

Course Title	**Data Warehousing and Data Mining**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	• Apply understanding of data warehousing concepts, including the architecture, organization, and functionality of data warehouses in business intelligence.
	• Analyze the requirements for a data warehouse schema, designing and implementing one while considering factors like dimension hierarchies.
	• Evaluate the importance of metadata in data warehousing, developing strategies for managing metadata to support data governance and data lineage.
	• Understand and apply various data mining algorithms such as classification, clustering, and association rule mining, recognizing their role in extracting insights from large datasets.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of the basic concepts of data warehousing, including its architecture, components, and its role in decision support systems.
	• Analyze the requirements for a data warehouse schema, designing and implementing one while considering factors like the star schema, snowflake schema, and dimension hierarchies. Star schema involves a central fact table and denormalized dimension tables, while snowflake schema involves normalized dimension tables. Dimension hierarchies represent relationships between different levels in a dimension.
Content Outline	• **Introduction:** Need for data warehousing, Basic elements of data warehousing, Trends in data warehousing. • **Project planning and management**, collecting the requirements. • Architectural Components, Infrastructure and metadata. • Principles of dimensional modeling, Dimensional modeling advanced topics, data extraction, transformation and loading, data Quality.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of Extract, Transform, and Load (ETL) processes, which involve extracting data from source systems, transforming it for analysis, and loading it into a data warehouse.
	• Analyze and integrate data from diverse sources into an organized and unified data warehouse, addressing issues related to data quality and consistency.
	• Evaluate and define data mining, understanding its role in extracting valuable patterns, trends, and insights from large datasets. Data mining techniques can include clustering, classification, regression, and association rule mining.
Content Outline	• **Information Access And Delivery:** Matching information to classes of users, OLAP in data warehouse, Data warehousing and the web. • **Implementation And Maintenance:** Physical design process, data warehouse deployment, growth and Maintenance.

	• **Data Mining:** Introduction Basics of data mining, related concepts, Data mining techniques. • **Data Mining Algorithms:** Classification, Clustering, Association rules.
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze web data to extract useful information, patterns, and knowledge.
	• Apply various data mining algorithms, such as clustering, classification, regression, association rule mining, and anomaly detection.
	• Evaluate data pre-processing strategies for mining, including handling missing values and transforming variables.
	• Design efficient data extraction and mining processes to better analyze and derive insights from data.
Content Outline	• **Knowledge Discovery:** KDD Process • **Web Mining:** Web Content Mining, Web Structure Mining, Web Usage mining. • **Advanced Topics:** Spatial mining, Temporal mining. Visualisation : Data generalization and summarization-based characterization, Analytical • **characterization :** analysis of attribute relevance, Mining class comparisons: Discriminating between different classes, Mining Descriptive statistical measure sing large databases
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze popular data mining tools and software to effectively implement and experiment with various algorithms and techniques.
	• Apply understanding of the different trends and applications in data mining in various fields.
	• Evaluate strategies for enhancing the performance of data warehouse queries, such as indexing and partitioning.
	• Design and implement effective strategies for optimal data warehouse query performance.
Content Outline	• **Data Mining Primitives, Languages, and System Architectures**: Data mining primitives, Query language, Designing GUI based on a data mining query language, Architectures of data mining systems • **Application and Trends in Data Mining:** Applications, Systems products and research prototypes, Additional themes in data mining, Trends in data mining

Assignments/Activities to wards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity: Conceptualize, design, and implement a simple data warehouse schema. Use a dataset of your choice, consider elements like the star schema, snowflake schema, and dimension hierarchies. Write a report explaining your design choices and how they relate to the basic concepts of data warehousing.

Module 2:

Activity: Select a dataset and perform an Extract, Transform, and Load (ETL) process on it. Document any issues related to data quality and consistency and how you addressed them. Then, apply a simple data mining technique (like classification or clustering) to extract patterns from your dataset.

Module 3:

Activity: Choose a set of web data and apply data mining algorithms such as clustering, classification, regression, association rule mining, and anomaly detection. Design a preprocessing strategy to handle missing values and transform variables. Extract and document any interesting patterns or insights you find.

Module 4:

Activity: Using a popular data mining tool of your choice, implement various algorithms and techniques on a chosen dataset. Evaluate the performance of these techniques and propose strategies for enhancing the performance of data warehouse queries. Write a brief report on current trends and applications in data mining in your field of interest.

References: -

1. Ponnian, Paulraj. Data Warehousing Fundamentals. John Wiley.
2. Dunham, M. H. Data Mining: Introductory and Advanced Topics. Pearson Education.
3. Han, Jiawei, and Micheline Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann. Kimball, Ralph. The Data Warehouse Lifecycle Toolkit. John Wiley.
4. Berry, Michael, and Gordon Linoff. Mastering Data Mining. John Wiley.
5. Inmon, W. H. Building the Data Warehouse. Wiley Dreamtech.
6. Mallach, E. G. Decision Support and Data Warehouse Systems. TMH.

2.5 Major (Elective-II)-B

Course Title	**Introduction To Data Science**
Course Credits	**4**
Course Outcomes	**After going through the course, learners will be able to**
	• Analyze different types and sources of data, along with the techniques to collect and interpret it.
	• Apply understanding of various statistical concepts like distributions, hypothesis testing, confidence intervals, and correlation.
	• Evaluate various machine learning algorithms such as Linear Regression, Logistic Regression, Decision Trees, Clustering, etc.
	• Design and implement text mining techniques to analyze unstructured data, possibly including hands-on lab sessions for practical understanding.
Module 1 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze various types of data, identifying and differentiating between them based on their characteristics and use cases.

	• Apply understanding of data analysis to construct models for predictive or descriptive purposes.
	• Evaluate and implement feature extraction methodologies to enhance model performance and interpretability.
	• Design illustrations to explain the stages within the Data Science lifecycle and their significance in problem-solving.
Content Outline	• **Introduction to Data :**What is Data? Different kinds of data, Data Sources, Different types of data sources, • Exploratory Data Analysis (EDA), Data Science lifecycle, Data Collection • Data Extraction, Data Analysis & Modeling • **Data transformations:** Dimension reduction, Feature extraction, Smoothing and aggregating.
Module 2 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze and perform various mathematical operations efficiently on NumPy arrays, a powerful tool for mathematical computation in Python.
	• Apply understanding of the data structures provided by Pandas (Series, DataFrame) and utilize them effectively in data analysis tasks.
	• Evaluate and execute merging operations to combine datasets efficiently for comprehensive analysis.
	• Design strategies to apply various types of joins effectively to merge datasets based on common columns or indices.
Content Outline	• **The World of arrays with Numpy:** creating an array, Mathematical operations, Indexing and slicing, Shape manipulation. • **Empowering Data analysis with pandas:** the data structure of pandas, Inserting and exporting data • **Data Cleansing:** checking missing data, filling missing data, merging operations • **Data Operations:** Aggregation operations, Joins
Module 3 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Analyze various probability distributions (normal, binomial, Poisson, etc.), identifying and differentiating between them based on their characteristics and real-world applications.
	• Apply control over line properties in various chart types for better visualization, enhancing the clarity and effectiveness of data representation.
	• Evaluate and differentiate between various types of Machine Learning (supervised, unsupervised, reinforcement learning), understanding their unique characteristics and applications.
Content Outline	• **Inferential Statistics:** Various forms of distribution, z-score, p-value, Type 1 and Type 2 errors, Confidence Interval, Correlation, Chi-square distribution, ANOVA • **Making Sense of Data Through Visualization:** Controlling the line properties of a chart, creating multiple plots, styling your plots, Boxplots, Heatmaps, Scatter plots with histogram, Bubble charts • **Uncovering Machine Learning:** Different types of Machine Learning, Linear Regression, Logistic Regression, Decision

	Tree, K-means Clustering, Hierarchical Clustering
Module 4 (Credit 1)	
Learning Outcomes	**After learning the module, learners will be able to**
	• Apply understanding of User-Based Collaborative Filtering techniques to generate recommendations based on user similarities.
	• Analyze and explain the theory behind text mining and its applications in analyzing unstructured data, providing clear and comprehensive insights.
Content Outline	• **Generating Recommendations Systems:** User Based collaborative filtering, Item Based collaborative filtering, Context Based filtering • **Case Study Theory:** Analyzing Unstructured Data using Text mining techniques. (Case Study Practical Implementation to be performed in lab as part of Practical's)

Assignments/Activities to wards Comprehensive Continuous Evaluation (CCE)

Module 1:

Activity: Conduct an exploratory data analysis on a dataset of your choice. Identify the different types of data present, apply your understanding of data analysis to construct predictive models, implement feature extraction methodologies to enhance model performance, and design illustrations to explain your process and findings.

Module 2:

Activity: Using Python, create a NumPy array and perform various mathematical operations on it. Then, use a dataset of your choice and apply your understanding of data structures using Pandas (Series, DataFrame) for data analysis tasks. Perform merging operations and apply various types of joins to combine datasets.

Module 3:

Activity: Analyze different probability distributions (normal, binomial, Poisson, etc.) and their real-world applications. Create different types of charts with controlled line properties for better visualization. Lastly, differentiate between various types of Machine Learning (supervised, unsupervised, reinforcement learning), and implement one of them on a dataset.

Module 4:

Activity: Implement a User-Based Collaborative Filtering technique to generate recommendations based on user similarities on a dataset. Explain the theory behind your approach. Also, find a set of unstructured data and apply text mining techniques to analyze it. Provide comprehensive insights from your analysis.

References: -

1. Provost, F., & Fawcett, T. (2013). Data Science for Business. O'Reilly Publication.
2. McKinney, W. (2017). Python for Data Analysis. O'Reilly Publication.

3. Newbold, P., & Carlson, W. L. (2012). Statistics for Business and Economics. Pearson Publication
4. Müller, A. C., & Guido, S. (2016). Introduction to Machine Learning with Python. O'Reilly Publication.
5. Madhavan, S. (2015). Mastering Python for Data Science: Explore the World of Data Science Through Python and Learn How to Make Sense of Data. Packt Publishing. Packt Publishing.
6. VanderPlas, J. (2016). Python Data Science Handbook: Essential Tools for Working with Data. O'Reilly.

2.6. OJT

Course Title	**On Job Training (OJT)**
Course Credits	**4 Credits**