Programme Structure

School of Engineering & Technology

B. Tech. CSE(Hons)

Programme Code: ET0101

Batch: 2023-2027

Programme: B. Tech.CSE(Hons)
AY: 25-26

Semester: V **Batch2023-27**

S. No	Status	Paper Code	Subjects	Study Scheme Lec / Week			Hours	Credits	CIE	ESE	Total	Pass Marks
				L	T	P						
1	CC-1	B010123501	Analyzing, Visualizing and Applying data science with Python	3	1	0	4	3	50	50	100	40
2	CC-2	B010123502	Software Engineering	3	0	0	3	3	50	50	100	40
3	CC-3	B010123503	Data Base Management Systems	4	0	0	4	4	50	50	100	40
4	CC-4	B010123504	Computer Networks	4	0	0	4	4	50	50	100	40
5	DSC(minor)-1	B010123505	Cloud Computing	4	0	0	4	4	50	50	100	40
6	CC-5	B010123551	Analyzing, Visualizing and Applying data science with Python Lab	0	0	2	2	1	60	40	100	40
7	CC-6	B010123552	Software Engineering Lab	0	0	2	2	1	60	40	100	40
8	CC-7	B010123553	DBMS Lab	0	0	2	2	1	60	40	100	40
9	CC-8	B010123554	Computer Networks Lab	0	0	2	2	1	60	40	100	40
10	Internship	B010123556	Internship Evaluation	0	0	2	2	1	100		100	40
			Total	18	1	10	29	23	590	410	1000	400

DETAILED 5th YEAR CURRICULUM CONTENTS

Undergraduate Degree in Engineering &Technology

PROGRAM: B.Tech. CSE(H)

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester-5th**
Course Name- Analyzing, Visualizing and Applying data science with Python			
A.Y 2025-26	**Course Code- B010123501**	**Batch-2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course - Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To introduce the fundamentals of Python programming and its environment for data science applications.
2. To equip students with data handling and preprocessing skills using NumPy and Pandas libraries.
3. To develop the ability to visualize and interpret data using Matplotlib, Seaborn, and Plotly.
4. To perform exploratory data analysis and apply basic statistical techniques on real-world datasets.
5. To understand core machine learning concepts and apply basic models for classification and prediction.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction to Python for Data Science:** Introduction to Data Science and Python, Python basics: Variables, Data types, Operators, Control structures: if-else, loops, Functions and Modules, Data structures: Lists, Tuples, Sets, Dictionaries, Working with files (read/write operations), Introduction to Jupyter Notebook and Google Colab.	09
2	**Data Handling and Processing with NumPy & Pandas** : Introduction to NumPy: Arrays, Operations, Indexing, Broadcasting, Introduction to Pandas: Series and DataFrames, Data cleaning: Handling missing data, duplicates, Data transformation: Filtering, Grouping, Sorting, Merging, Handling time series data, Reading and writing data: CSV, Excel, JSON.	09
3	**Data Visualization with Matplotlib, Seaborn & Plotly:** Introduction to data visualization, Basic plotting with Matplotlib: line, bar, scatter, histogram, Advanced visualizations with Seaborn: heatmaps, boxplots, pair plots, Interactive visualizations using Plotly, Best practices for visualization: labeling, scaling, aesthetics, Dashboard basics using Plotly Dash.	09
4	**Exploratory Data Analysis (EDA) and Statistics:** Descriptive statistics: mean, median, mode, standard deviation, Data distributions and outliers, Correlation and covariance, Data normalization and scaling, Using Pandas Profiling and Sweetviz for automated EDA, Case study: EDA on a real-world dataset.	09
5	**Introduction to Machine Learning Applications:** Introduction to Machine Learning: Supervised vs Unsupervised learning, Data preprocessing: Encoding, scaling, train-test split, Basic models: Linear Regression, Logistic Regression, KNN, Decision Trees, Model evaluation: Accuracy, Precision, Recall, Confusion Matrix, Real-world application: Predictive modeling on sample datasets, Overview of end-to-end data science workflow.	09

Course Outcomes

CO 1	Basic Python programs and apply fundamental programming constructs for data manipulation and control flow.
CO 2	Manipulate and preprocess structured datasets using NumPy arrays and Pandas DataFrames for data analysis tasks.
CO 3	Static and interactive data visualizations using Python libraries to effectively communicate patterns and insights.
CO 4	Exploratory data analysis and statistical techniques to uncover trends, detect outliers, and derive summary insights from data.
CO5	Basic machine learning models using Scikit-learn to perform predictive analysis and evaluate model performance.

Text books:

1. Data Visualization with Python and JavaScript, Kyran Dale, Shroff Publisher/O'Reilly Publisher Publication.
2. Data Science Using Python and R by Chantal D. Larose and Daniel T. Larose, Wiley Publication.
3. Practical Statistics for Data Scientists" by Peter Bruce & Andrew Bruce.
4. *"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron*

Reference Books:

5. Python for Data Science and Visualization -Beginners to Pro, Udemy.
6. *"Python for Everybody: Exploring Data in Python 3"* by Charles Severance.
7. "Python for Data Analysis" by Wes McKinney.

School Name- **School of Engineering & Technology**			
Program- B. Tech.CSE(Hons)			**Semester-5th**
Course Name - Software Engineering			
A.Y 2025-26	**Course Code- B010123502**	**Batch-2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the fundamental concepts of software engineering, its processes, and various software development life cycle (SDLC) models.
2. To analyze and develop Software Requirement Specifications (SRS) using modeling tools and IEEE standards.
3. To explore software design principles, architectural and modular design techniques, and software quality assurance frameworks.
4. To implement effective software testing strategies including white-box and black-box testing, static testing, and debugging techniques.
5. To understand software maintenance, configuration management, cost estimation models (e.g., COCOMO), and project risk analysis.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	**Introduction:** Introduction to Software Engineering, Software Components, SoftwareCharacteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Quality Attributes. Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.	**09**
2	**Software Requirement Specifications (SRS):** Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modelling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, IEEE Standards for SRS. Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks, ISO 9000 Models, SEI-CMM Model.	**09**
3	**Software Design:** Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design. Software Measurement and Metrics: Various Size Oriented Measures: Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.	**09**
4	**Software Testing:** Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, TopDown and Bottom- Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies: Formal Technical Reviews (Peer Reviews), Walk Through, Code Inspection, Compliance	**09**

	with Design and Coding Standards.	
5	**Software Maintenance and Software Project Management:** Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.	**09**

Course Outcomes

CO 1	Explain various software characteristics and analyze different software DevelopmentModels
CO 2	Demonstrate the contents of a SRS and apply basic software quality assurance practices toensure that design, development meet or exceed applicable standards
CO 3	Compare and contrast various methods for software design.
CO 4	Formulate testing strategy for software systems, employ techniques such as unit testing, Testdriven development and functional testing
CO 5	Manage software development process independently as well as in teams and make use ofVarious software management tools for development, maintenance and analysis.

Text books:

1. RS Pressman, Software Engineering: A Practitioners Approach, McGraw Hill.
2. Pankaj Jalote, Software Engineering, Wiley
3. Rajib Mall, Fundamentals of Software Engineering, PHI Publication.
4. KK Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.

Reference Books:

5. Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication.
6. Ian Sommerville, Software Engineering, Addison Wesley.
7. Kassem Saleh, “Software Engineering”, Cengage Learning.
8. P fleeger, Software Engineering, Macmillan Publication

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester-5th**
Course Name- DataBase Management Systems			
A.Y 2025-26	**Course Code- B010123503**	**Batch-2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50 hrs	**Total Credits- 04**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the fundamental concepts of database systems, architecture, and data models including ER and EER models.
2. To explore relational database concepts, query languages such as SQL, and relational algebra/calculus for data manipulation.
3. To design normalized relational schemas using functional dependencies, normal forms, and various decomposition techniques.
4. To study transaction processing, concurrency control, and recovery mechanisms to ensure database consistency and reliability.
5. To examine advanced topics like PL/SQL, triggers, stored procedures, and distributed database systems with concurrency strategies.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction:** Overview, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence and Database Language and Interfaces, Data Definitions Language, DML, Overall Database Structure. Data Modeling Using the Entity Relationship Model: ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Extended ER Model, Relationship of Higher Degree.	**10**
2	**Relational data Model and Language:** Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra, Relational Calculus, Tuple and Domain Calculus. Introduction on SQL: Characteristics of SQL, Advantage of SQL. SQl Data Type and Literals. Types of SQL Commands. SQL Operators and Their Procedure. Tables, Views and Indexes. Queries and Sub Queries. Aggregate Functions. Insert, Update and Delete Operations, Joins, Unions, Intersection, Minus, Cursors, Triggers, Procedures in SQL/PL SQL	**10**
3	**Data Base Design & Normalization:** Functional dependencies, normal forms, first, second, & third normal forms, BCNF, inclusion dependence, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design	**10**
4	**Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Checkpoints, Deadlock Handling. Distributed Database: Distributed	**10**

	Data Storage, Concurrency Control, Directory System.	
5	**Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Multi Version Schemes, Recovery with Concurrent Transaction, Case Study of Oracle.	**10**

Course Outcomes

CO 1 Apply knowledge of database for real life applications.
CO 2 Apply query processing techniques to automate the real time problems of databases.
CO 3 Identify and solve the redundancy problem in database tables using normalization.
CO 4 Understand the concepts of transactions, their processing so they will familiar with broad range of database management issues including data integrity, security and recovery.
CO 5 Design, develop and implement a small database project using database tools.

Text books:

1. Korth, Silbertz, Sudarshan," Database Concepts", McGraw Hill
2. Date C J, "An Introduction to Database Systems", Addision Wesley
3. Elmasri, Navathe, " Fundamentals of Database Systems", Addision Wesley
4. O'Neil, Databases, Elsevier Pub.

Reference Books:

5. RAMAKRISHNAN"Database Management Systems",McGraw Hill
6. Leon &Leon,"Database Management Systems", Vikas Publishing House
7. Bipin C. Desai, " An Introduction to Database Systems", Gagotia Publications
8. Majumdar& Bhattacharya, "Database Management System", TMH

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester-5th**
Course Name- Computer Networks			
A.Y 2025-26	**Course Code- B010123504**	**Batch-2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50 hrs	**Total Credits- 04**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the goals, architecture, and models of computer networks, including OSI and TCP/IP protocols.
2. To explore physical layer concepts such as transmission media, topologies, encoding, and network performance.
3. To study link layer mechanisms including framing, error control, flow control, and medium access protocols.
4. To analyze network layer functionalities like addressing, routing algorithms, congestion control, and IPv6.
5. To examine transport and application layer protocols for end-to-end communication, congestion control, and network services.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introductory Concepts**: Goals and applications of networks, Categories of networks, Organization of the Internet, ISP, Network structure and architecture (layering principles, services, protocols and standards), The OSI reference model, TCP/IP protocol suite, Network devices and components. **Physical Layer:** Network topology design, Types of connections, Transmission media, Signal transmission and encoding, Network performance and transmission impairments, Switching techniques and multiplexing.	10
2	**Link layer:** Framing, Error Detection and Correction, Flow control (Elementary Data Link Protocols, Sliding Window protocols). Medium Access Control and Local Area Networks: Channel allocation, Multiple access protocols, LAN standards, Link layer switches & bridges (learning bridge and spanning tree algorithms).	10
3	**Network Layer:** Point-to-point networks, Logical addressing, Basic internetworking (IP, CIDR, ARP, RARP, DHCP, ICMP), Routing, forwarding and delivery, Static and dynamic routing, Routing algorithms and protocols, Congestion control algorithms, IPv6.	10
4	**Transport Layer:** Process-to-process delivery, Transport layer protocols (UDP and TCP), Multiplexing, Connection management, Flow control and retransmission, Window management, TCP Congestion control, Quality of service.	10
5	**Application Layer:** Domain Name System, World Wide Web and Hyper Text Transfer Protocol, Electronic mail, File Transfer Protocol, Remote login, Network management, Data	10

	compression, Cryptography – basic concepts.	

Course Outcomes

CO1	Explain basic concepts, OSI reference model, services and role of each layer of OSI model and TCP/IP, networks devices and transmission media, Analog and digital data transmission.
CO2	Apply channel allocation, framing, error and flow control techniques.
CO3	Describe the functions of Network Layer i.e. Logical addressing, subnetting& Routing Mechanism.
CO4	Explain the different Transport Layer function i.e. Port addressing, Connection Management, Error control and Flow control mechanism.
CO5	Explain the functions offered by session and presentation layer and their Implementation.

Text books:

1. Behrouz Forouzan, "Data Communication and Networking", McGraw Hill
2. Andrew Tanenbaum "Computer Networks", Prentice Hall.
3. William Stallings, "Data and Computer Communication", Pearson.

Reference Books:

4. Kurose and Ross, "Computer Networking- A Top-Down Approach", Pearson.
5. Peterson and Davie, "Computer Networks: A Systems Approach", Morgan Kaufmann
6. W. A. Shay, "Understanding Communications and Networks", Cengage Learning.
7. D. Comer, "Computer Networks and Internets", Pearson.
8. Behrouz Forouzan, "TCP/IP Protocol Suite", McGraw Hill.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester-5th**
Course Name- Cloud Computing			
A.Y 2025-26	**Course Code- B010123505**	**Batch-2023-27**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50 hrs	**Total Credits- 04**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the fundamentals of descriptive statistics including data visualization, central tendency, dispersion, and correlation.
2. To learn the principles of probability theory, probability distributions, and foundational theorems for statistical inference.
3. To apply inferential statistical methods such as hypothesis testing, confidence intervals, and regression analysis.
4. To explore concepts in vector spaces, inner product spaces, and orthonormal basis for mathematical modeling.
5. To analyze linear transformations, eigenvalues, and matrix diagonalization for solving advanced algebraic problems.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	**Introduction:** Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed, History of Cloud Computing - Cloud Architecture - Types of Clouds - Business models around Clouds – Major Players in Cloud Computing- issues in Clouds - Eucalyptus - Nimbus - Open Nebula, CloudSim	**10**
2	**Cloud Services:** Types of Cloud services: Software as a Service- Platform as a Service – Infrastructure as a Service - Database as a Service - Monitoring as a Service – Communication as services. Service providers- Google, Amazon, Microsoft Azure, IBM, Sales force.	**10**
3	**Collaborating Using Cloud Services:** Email Communication over the Cloud - CRM Management – Project Management-Event Management - Task Management – Calendar - Schedules - Word Processing – Presentation – Spreadsheet - Databases – Desktop - Social Networks and Groupware.	**10**
4	**Virtualization for Cloud:** Need for Virtualization – Pros and cons of Virtualization – Types of Virtualization –System VM, Process VM, Virtual Machine monitor – Virtual machine properties - Interpretation and binary translation, HLL VM - supervisors – Xen, KVM, VMware, Virtual Box, Hyper-V.	**10**
5	**Security, Standards and Applications:** Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed management Task Force – Standards for application Developers – Standards for Messaging – Standards for Security, End user access to cloud computing, Mobile Internet devices and the cloud. Hadoop – MapReduce – Virtual Box — Google App Engine – Programming Environment for Google App Engine	**10**

Course Outcomes

CO1 Explain the foundational concepts of cloud computing.

CO2	Identify and differentiate among various cloud service models (SaaS, PaaS, IaaS, etc.) and analyze offerings from major cloud service providers.
CO3	Demonstrate effective use of cloud-based collaborative tools and applications for communication, project management, productivity, and social networking.
CO4	Evaluate the role of virtualization in cloud computing by comparing virtualization types, virtual machine monitors, and platforms such as Xen, VMware, KVM, and Hyper-V.
CO5	Assess security issues and standardization in cloud environments.

Text books:

1. David E.Y. Sarna, "Implementing and Developing Cloud Application", CRC press 2011.
2. Lee Badger, Tim Grance, Robert Patt-Corner, Jeff Voas, NIST, Draft cloud computing synopsis and recommendation, May 2011.
3. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", McGrawHill 2010.
4. Haley Beard, "Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs", Emereo Pty Limited, July 2008.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester- 5th**
Course Name: Software Engineering Lab			
A.Y 2025-26	**Course Code- B010123552**	**Batch-2023-27**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. Apply inbuilt and custom conditional formatting and advanced lookup functions (VLOOKUP, HLOOKUP, XLOOKUP) to perform dynamic data highlighting and real-time search.
2. Use OFFSET and MATCH functions with SUM, AVERAGE, and COUNTIF to create dynamic ranges and analytical reports in Excel.
3. Utilize Excel cloud capabilities including report editing, exporting, embedding, permission management, and automated mail subscriptions.
4. Design interactive HR dashboards using functions like INDEX-MATCH, SUMIF, AVERAGEIF, COUNTIF, and dropdowns for real-time analytics.
5. Build financial models and perform business analytics through financial functions (PMT, IRR, XIRR), forecasting, descriptive/predictive statistics, Power Query, macros, and VBA automation.

	Topics	**No. of Teaching hours/ (Lecture)**
	For any given case/ problem statement do the following; 1. Prepare a SRS document in line with the IEEE recommended standards. 2. Draw the use case diagram and specify the role of each of the actors. Also state the precondition, postcondition and function of each use case. 3. Draw the activity diagram. 4. Identify the classes. Classify them as weak and strong classes and draw the class diagram. 5. Draw the sequence diagram for any two scenarios. 6. Draw the collaboration diagram. 7. Draw the state chart diagram. 8. Draw the component diagram. 9. Perform forward engineering in java. (Model to code conversion) 10. Perform reverse engineering in java. (Code to Model conversion) 11. Draw the deployment diagram.	**15**

Course Outcomes

CO 1 To understand the concept of look-up functions in Excel.
CO2 To gain the practical knowledge of worksheet.
CO3 To gain the practical knowledge of Excel Function & Formulas.
CO4 To gain the practical knowledge of Financial modelling in Excel.
CO5 To gain the practical knowledge of Business Analytics in Excel.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester- 5th**
Course Name: DBMS Lab			
A.Y 2025-26	**Course Code- B010123553**	**Batch-2023-27**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. Implement and analyze various sorting algorithms such as Bubble, Insertion, Selection, Shell, Radix, and Quick Sort for efficient data organization.
2. Develop stack-based applications using arrays and linked lists including postfix evaluation, parenthesis balancing, and infix-to-postfix conversion.
3. Construct and manipulate different types of queues including ordinary, circular, priority, and double-ended queues using array and linked list implementations.
4. Create and operate on various types of linked lists (singly, circular, doubly) to perform insertion, deletion, traversal, searching, counting, reversing, and polynomial operations.
5. Apply tree and graph data structures including binary search trees, heaps, expression trees, and graph algorithms like BFS and MST (using Kruskal's/Prim's methods) for solving real-world problems.

UNIT-1	**Topics** **List of Experiments (Indicative & not limited to)**	**No. of Teaching hours/ (Lecture)**
	1. Installing oracle/ MYSQL 2. Creating Entity-Relationship Diagram using case tools. 3. Writing SQL statements Using ORACLE /MYSQL: a) Writing basic SQL SELECT statements. b) Restricting and sorting data. c) Displaying data from multiple tables. d) Aggregating data using group function. e) Manipulating data. f) Creating and managing tables. 4. Normalization 5. Creating cursor 6. Creating procedure and functions 7. Creating packages and triggers	**15**

Course Outcomes

CO1: Apply and implement various sorting algorithms including Bubble, Insertion, Selection, Shell, Radix, and Quick sort to analyze time complexity and improve data organization.

CO2: Demonstrate proficiency in searching and hashing techniques, including linear/binary search, hashing methods, and collision resolution strategies to optimize data retrieval.

CO3: Design and implement stack-based operations and algorithms such as postfix evaluation, infix to postfix conversion, and parenthesis balancing using array and linked list representations.

CO4: Construct various queue structures — circular, priority, and double-ended queues — using array and linked list implementations to manage sequential data efficiently.

CO5: Develop and manipulate dynamic data structures including linked lists, trees (BST, Heaps), and graphs, and apply traversal, searching, and spanning tree algorithms for advanced data operations.

]

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester- 5th**
Course Name: Computer Networks Lab			
A.Y 2025-26	**Course Code- B010123554**	**Batch-2023-27**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. Understand and implement basic data link layer protocols like Stop-and-Wait and Sliding Window for reliable communication.
2. Learn socket programming concepts and develop client-server applications to facilitate network interactions.
3. Gain hands-on experience in simulating network protocols such as ARP, RARP, PING, and TRACEROUTE for network diagnostics.
4. Develop network applications using TCP and UDP sockets for real-world services including HTTP, RPC, chat, and file transfer.
5. Explore network simulation tools to study congestion control and analyze network performance under various scenarios.

	Topics **List of Experiments (Indicative & not limited to)**	**No. of Teaching hours/ (Lecture)**
	1. Implementation of Stop and Wait Protocol and Sliding Window Protocol. 2. Study of Socket Programming and Client – Server model 3. Write a code simulating ARP /RARP protocols. 4. Write a code simulating PING and TRACEROUTE commands 5. Create a socket for HTTP for web page upload and download. 6. Write a program to implement RPC (Remote Procedure Call) 7. Implementation of Subnetting . 8. Applications using TCP Sockets like a. Echo client and echo server b. Chat c. File Transfer 9. Applications using TCP and UDP Sockets like d. DNS e. SNMP f. File Transfer 10. Study of Network simulator (NS).and Simulation of Congestion Control Algorithms using NS.	**15**

Course Outcomes

CO1: Implement and analyze Stop-and-Wait and Sliding Window protocols for reliable data transmission.
CO2: Develop client-server applications using socket programming to enable network communication.
CO3: Simulate key network protocols such as ARP, RARP, PING, and TRACEROUTE for practical understanding.

CO4: Design and implement TCP and UDP-based applications including HTTP, RPC, chat, and file transfer.
CO5: Utilize network simulators to study congestion control algorithms and analyze network behavior.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons)			**Semester- 5th**
Course Name : **Analyzing, Visualizing and Applying data science with Python Lab**			
A.Y 2025-26	**Course Code- B010123551**	**Batch-2023-27**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. Set up Python environment and learn essential programming concepts for data analysis and machine learning.
2. Apply mathematical and scientific computing techniques using libraries like NumPy and SciPy for problem-solving.
3. Manipulate and analyze complex datasets efficiently using the Pandas library.
4. Build predictive models and perform machine learning tasks using Scikit-Learn on real-world datasets.
5. Develop practical applications such as employee attrition prediction, customer clustering**,** recommendation systems, and pattern recognition using Python tools and algorithms.

	Topics **List of Experiments (Indicative & not limited to)**	**No. of Teaching hours/ (Lecture)**
	1. Python Environment setup and Essentials. 2. Mathematical computing with Python (NumPy). 3. Scientific Computing with Python (SciPy). 4. Data Manipulation with Pandas. 5. Prediction using Scikit-Learn 6. Use python to predict employee attrition in a firm and help them plan their manpower. (take data set from kaggle). 7. Create customer clusters using different market strategies on a data set. 8. Make a movie recommendation system. 9. Develop a prediction mechanism to predict which employee can go on leave in a company in near future. 10. Recognizing alphabets using SVM. 11. Simulation of Congestion Control Algorithms using NS.	**15**

Course Outcomes

CO1 Set up and utilize the Python programming environment along with essential libraries for data science.
CO2 Apply NumPy and SciPy for efficient mathematical and scientific computing tasks.
CO3 Perform data manipulation and analysis using the Pandas library to handle real-world datasets.
CO4 Build and evaluate predictive models using Scikit-Learn for classification and regression problems.
CO5 Implement machine learning solutions such as clustering, recommendation systems, and SVM- based classification on practical datasets.

CO 4	Improving problem-solving, critical thinking skills and report writing.
CO 5	Learning professional skills like exercising leadership, behaving professionally, behaving ethically, listening effectively, participating as a member of a team, developing appropriate workplace attitudes