



**Semester-wise Details of the Courses**

**SEMESTER -1**

Catalog Course Code	Course Code	Course Type	Course Title	Engagement Hrs/Week(Teaching Scheme)				Credits	Continuous Assessment		Term End Examination		Total
				L	T	P	Total		Theory	Practical	Theory	Practical	
STENCS 057V0	DP04020 125101	PCC	Advances in Algorithms	3	-	-	3	3	90	0	60	0	150
STENCS 058V0	DP04020 125102	PCC	Advanced Database Management Systems	3	-	-	3	3	90	0	60	0	150
STENCS 059V0	DP04020 125103	PCC	Research Methodology & IPR	3	-	-	3	3	90	0	60	0	150
STENCS 060V0	DP04020 125104	PCC	Advances in Algorithms Lab	-	-	2	2	1	0	50	0	0	50
STENCS 061V0	DP04020 125105	PCC	Advanced Database Management Systems Lab	-	-	2	2	1	0	0	0	50	50
<b>Total</b>				<b>9</b>	<b>-</b>	<b>4</b>	<b>13</b>	<b>11</b>	<b>270</b>	<b>50</b>	<b>180</b>	<b>50</b>	<b>550</b>
<b>Program Elective-I (Choose any one)</b>													
STENCS 062V0	DP04020 125106	PEC	Advanced Data Science and Analytics	3	-	-	3	3	90	0	60	0	150
STENCS 063V0	DP04020 125107	PEC	Cryptography and Network Security	3	-	-	3	3	90	0	60	0	150
STENCS 064V0	DP04020 125108	PEC	Distributed Systems	3	-	-	3	3	90	0	60	0	150
STENCS 065V0	DP04020 125109	PEC	Human Computer Interface	3	-	-	3	3	90	0	60	0	150
STENCS 066V0	DP04020 125110	PEC	Advanced Web Technology	3	-	-	3	3	90	0	60	0	150
STENCS 067V0	DP04020 125111	PEC	Digital Image Processing	3	-	-	3	3	90	0	60	0	150
STENCS 068V0	DP04020 125112	PEC	Parallel Algorithms	3	-	-	3	3	90	0	60	0	150
<b>Total</b>				<b>3</b>	<b>-</b>	<b>-</b>	<b>3</b>	<b>3</b>	<b>90</b>	<b>0</b>	<b>60</b>	<b>0</b>	<b>150</b>
STENCS 069V0	DP04020 125113	VSEC	Software & Tools for Scientific Research-I	-	-	4	4	2	0	100	0	0	100
STENCS 070V0	DP04020 125114	VEC	Environmental Study	1	1	-	2	2	50	50	0	0	100
HLHMC S005V0	DP04020 125115	MNC	Yoga and Fitness	-	-	4	4	2	0	100	0	0	100
<b>Total Academic Engagement and Credits</b>				<b>13</b>	<b>01</b>	<b>12</b>	<b>26</b>	<b>20</b>	<b>410</b>	<b>300</b>	<b>240</b>	<b>50</b>	<b>1000</b>

DPGU

Note:

1 credit= 50 marks

L: Lecture, T: Tutorial, P: Practical



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

**Master of Technology (Computer Science and Engineering)**

**Program Structure Summary**

<b>Year</b>	<b>Semester</b>	<b>Credit Points</b>	<b>Marks</b>
1	1	20	1000
	2	20	1000
2	3	20	1000
	4	20	1000
<b>Total</b>		<b>80</b>	<b>4000</b>



**Dnyaan Prasad Global University**  
 Dr. D. Y. Patil Unitech Society  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Advances in Algorithms			<b>Code</b>	DP04020125101			
	Teaching Scheme (Hrs/Week)			Examination Scheme & Marks				
<b>Credits</b>	Lectures	Tutorials	Practical	CA		TEA		Total
				Theory	Practical	Theory	Practical	
<b>3</b>	3	--	-	90	--	60	--	150

**Prerequisite:**

- 1) Discrete Mathematics
- 2) Data Structure and Algorithms
- 3) Theory of Computation
- 4) Design and Analysis of Algorithms

**Course Objectives:**

1. To apply various algorithmic strategies in solving computational problems effectively.
2. To evaluate and compare the performance of algorithmic approaches based on time and space complexity.
3. To design algorithms that are optimized for both time and space efficiency.
4. To understand the principles and implementation of parallel and distributed algorithms.

**Course Outcomes:**

**On completion of the course, students will be able to:**

- CO1: Formulate** computational problems and **develop** appropriate algorithmic solutions to Classify P and NP complexity classes.
- CO2: Identify** suitable algorithmic strategies and **apply** them effectively to solve the given problem.
- CO3: Apply** Algorithmic Strategies to solve complex computational problems and **evaluate** their efficiency and correctness.

<p><b>CO4: Implement</b> different strategies to solve the problem optimally.</p> <p><b>CO5: Understand and Analyse</b> graph flow and matching algorithms to solve real time problems.</p> <p><b>CO6: Understand and apply</b> algorithmic solutions to real-life problems with Implementation and optimization in distributed and parallel environments.</p>		
<p><b>Course Contents</b></p>		
<p><b>Unit I</b></p>	<p><b>Algorithms and Complexity Theory</b></p>	<p><b>6 Hours</b></p>
<p>Algorithms: Role of Algorithms in computing, Design and Correctness of Algorithms, Growth of Functions: Asymptotic notations, Recurrences and Solution of Recurrence equations-The substitution method, The recurrence tree method, The master method, Polynomial and Non-Polynomial Problems: P Class of problems, NP Class of Problems, NP Hard problems and NP Complete problems. Approximation Algorithms: Vertex cover problem, The travelling Salesman problem, Randomized algorithms.</p>		
<p><b>Case Study#1:</b></p>	<p><b>Analysing and Solving the Vertex Cover Problem Using Approximation and Complexity Concepts.</b></p> <p>Consider an Undirected graph consists of vertices and edges (V, E), find the smallest subset of vertices such that every edge E is incident to at least one vertex in smallest subset of vertices.</p>	
<p><b>Case Study#2:</b></p>	<p><b>Optimizing AI Model Training with NP-Complete Constraints and Amortized Resource Management</b></p> <p>Consider training deep learning models (e.g., transformers) on resource-constrained hardware (like GPUs). Suppose the scheduling of model training tasks (layers, batches) on limited GPUs is constrained by NP-Complete problems (e.g., bin-packing, scheduling). We also analyse the growth of computation time as model size n increases</p>	
<p><b>Unit II</b></p>	<p><b>Greedy and Dynamic Programming Strategy</b></p>	<p><b>6 Hours</b></p>
<p><b>Greedy Strategy:</b> Control Abstraction, Activity Selection Problem, Fractional Knapsack, Job Sequencing with Deadlines. Minimum Spanning Trees: Prim’s and Kruskal’s Algorithms, Dijkstra’s Algorithm.</p> <p><b>Dynamic Programming:</b> Control Abstraction, 0/1 Knapsack, Longest Common Subsequence (LCS) and Longest Increasing Subsequence (LIS), Matrix Chain Multiplication, Optimal Binary Search Trees(OBST), Edit Distance and Variants (e.g., Levenshtein distance), Floyd-Warshall Algorithm (All-pairs shortest path)</p>		

<b>Case Study#3:</b>	<b>Optimize job scheduling and network routing for a logistics company aiming to maximize profits and minimize delivery time.</b> Consider any real time scenario of E-commerce delivery optimization system.	
<b>Unit III</b>	<b>Algorithmic Strategies</b>	<b>6 Hours</b>
<p><b>Backtracking:</b> Control Abstraction, Graph Colouring Problem, Constraint Satisfaction Problems, Branch and Bound.  <b>Probabilistic and Randomized Algorithms:</b> Probabilistic Analysis, Indicator Random Variables, Hiring Problem  Classes of Randomized Algorithm, <b>Randomized Algorithms:</b> Monte Carlo, Las Vegas, Linear Programming, Randomized Quick Sort.</p>		
<b>Case Study#4:</b>	<b>Efficient Hiring Process Using Randomized Algorithms</b> This case study explores the Hiring Problem, demonstrating how randomized algorithms and indicator random variables help optimize the expected quality of candidate selection while minimizing interviewing costs. The study applies probabilistic analysis to design hiring strategies that improve decision-making under uncertainty, showcasing practical benefits in recruitment and resource management.	
<b>Unit IV</b>	<b>Abstract Algorithms</b>	<b>6 Hours</b>
<p><b>Natural Algorithms</b> –Evolutionary Algorithms and Evolutionary Computing, Introduction to Genetic Algorithm, Simulated Annealing, Artificial Neural Network. <b>Optimization Techniques:</b> Newton’s Method and Quasi-Newton Methods, <b>Optimization Algorithms:</b> Swarm Optimization, Ant Colony Optimization.</p>		
<b>Case Study#5:</b>	<b>Traffic signal optimization and control using Genetic and Optimization Algorithms.</b>	
<b>Unit V</b>	<b>Graph Flows and Matching Algorithms</b>	<b>6 Hours</b>
<p>Flow Networks Basics, <b>Maximum Flow Algorithms:</b> Ford-Fulkerson Algorithm, Minimum Cut and MaxFlow Min-Cut Theorem, Applications.  <b>String Matching Algorithms:</b> Naïve string Matching; String matching with finite automata; Rabin - Karp algorithm, Knuth-Morris Pratt algorithm, Boyer-Moore algorithms. <b>Approximate String Matching:</b> Dynamic Programming for approximate matches.</p>		
<b>Case Study#6:</b>	<b>Text indexing in search engines</b>	
<b>Case Study#7:</b>	<b>DNA Sequence Analysis Using String-Matching Algorithms</b> This case study focuses on using advanced <b>string-matching algorithms</b> like, <b>BoyerMoore</b> , and <b>Rabin-Karp</b> to search for specific gene patterns or motifs within largescale DNA sequences. By efficiently comparing genomic data, these algorithms help identify critical biological markers, advancing research in genetics, medicine, and bioinformatics.	

Unit VI	Parallel and Distributed Algorithms	6 Hours
<p>Introduction to Parallel and Distributed Algorithms, <b>Parallel Computing Basics</b>: Performance Measures of Parallel Algorithms, Parallel Merge Sort, Parallel Bitonic Sort, Parallel Breadth First Search.</p> <p><b>Distributed Graph Algorithms</b>: Distributed Depth First Search, Distributed Spanning Tree Algorithm, Karger's Min-Cut Algorithm (Randomized), <b>Distributed Coordination Algorithms</b>: Distributed Mutual Exclusion (Ricart-Agrawala Algorithm), Distributed Deadlock Detection.</p>		
<b>Case Study#8:</b>	Use any Parallel and Distributed Algorithm for Ride Sharing platform to optimize routes and match riders and drives dynamically based on location	
<b>Total Hours: 36</b>		
<b>Text Books:</b>		
Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press; ISBN 978-0-262-03384-8.		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Horowitz and Sahani, "Fundamentals of Computer Algorithms", University Press, ISBN: 978 817371 6126, 81 7371 61262.</li> <li>2. Aho, Hopcroft, Ullman "The Design and Analysis of Computer Algorithms".</li> <li>3. G. Brassard, P. Bratley, "Fundamentals of Algorithmics", PHI</li> </ol>		
<b>E-Books/Online Learning Resources:</b>		
<ol style="list-style-type: none"> <li>1. <a href="https://homel.vsb.cz/~fai0013/Kniha_Algoritmy.pdf">https://homel.vsb.cz/~fai0013/Kniha_Algoritmy.pdf</a></li> <li>2. <a href="https://www.phindia.com/Books/BookDetail/9788120352124/parallel-and-distributed-computing-basu?srsltid=AfmBOopWwGRLJnpIVcAEoi65RkYe4fcQBYD6CRDJcFtC17IQwIHTHWF9">https://www.phindia.com/Books/BookDetail/9788120352124/parallel-and-distributed-computing-basu?srsltid=AfmBOopWwGRLJnpIVcAEoi65RkYe4fcQBYD6CRDJcFtC17IQwIHTHWF9</a></li> </ol>		
<b>Online Courses:</b>		
<ol style="list-style-type: none"> <li>1. <b>Design and Analysis of Algorithm</b> <ol style="list-style-type: none"> <li>a. Instructor : Madhavan Mukund</li> <li>b. Cover : Design and Analysis of Algorithm</li> <li>c. Link : <a href="https://nptel.ac.in/courses/106106131">https://nptel.ac.in/courses/106106131</a></li> </ol> </li> <li>2. <b>Design and Analysis of Algorithm</b> <ol style="list-style-type: none"> <li>a. Instructor : Bhanu Priya</li> <li>b. Cover : Algorithm Introduction</li> <li>c. Link : <a href="https://youtu.be/GQNT0v5zKhE?si=IAvx3r3me1EQZrSD">https://youtu.be/GQNT0v5zKhE?si=IAvx3r3me1EQZrSD</a></li> </ol> </li> <li>3. <b>Programming, Data Structures and Algorithms</b> <ol style="list-style-type: none"> <li>a. Instructor : N. S. Narayanaswamy</li> <li>b. Cover : Programming, Data Structures and Algorithms</li> <li>c. Link : <a href="https://youtu.be/IW-1-TOZsMw?si=SFo-gmTTOiLUBbOs">https://youtu.be/IW-1-TOZsMw?si=SFo-gmTTOiLUBbOs</a></li> </ol> </li> <li>4. <b>Graph Searching, BFS and DFS</b> <ol style="list-style-type: none"> <li>a. Instructor : Prof. Benny George K and Prof. Sajith Gopalan</li> <li>b. Cover : Graph Searching, BFS and DFS</li> <li>c. Link : <a href="https://youtu.be/u2moDPvDmac?si=b3ffPaX9fh4wMzMd">https://youtu.be/u2moDPvDmac?si=b3ffPaX9fh4wMzMd</a></li> </ol> </li> </ol>		

The CO-PO Mapping Matrix											
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	1	2	2	-	-	-	-	-	-
CO2	3	3	2	1	2	-	-	-	-	-	-
CO3	3	3	2	2	3	-	-	-	-	-	-
CO4	3	2	3	1	3	-	-	-	-	-	-
CO5	3	3	2	1	3	-	-	-	-	-	-
CO6	3	3	3	2	3	-	-	-	-	-	1



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Advanced Database Management Systems			<b>Code</b>	DP04020125102			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
4	3	--	--	90	--	60	--	150
<b>Prerequisite:</b>								
a) Database Management Systems								
<b>Other Courses in this Series:</b>								
a) Database Management Systems Laboratory								
<b>Course Objectives:</b>								
a) To provide a deep understanding of advanced database concepts such as query optimization, indexing techniques, and transaction management for efficient data handling. b) To explore the architecture and functioning of distributed databases, enabling students to design systems that support data distribution and replication across multiple nodes. c) To introduce emerging database technologies, including NoSQL, NewSQL, and columnar databases, for managing semi-structured and unstructured data. d) To study the techniques of query processing and optimization, helping students understand how complex SQL queries are parsed, interpreted, and optimized by the database engine. e) To develop the ability to analyze and implement concurrency control and recovery mechanisms, ensuring database consistency and reliability in multi-user environments. f) To enable students to work with real-world data systems, including hands-on experience with advanced DBMS tools, big data platforms, and modern database APIs.								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
a) Analyse and optimize complex SQL queries using advanced query processing techniques and cost-based optimization strategies								

- b) Apply appropriate indexing and hashing techniques to improve database query performance and retrieval speed.
- c) Design and implement distributed database systems with concepts of data fragmentation, replication, and distributed query processing.
- d) Evaluate concurrency control mechanisms and recovery strategies to ensure database consistency and fault tolerance in multi-user environments.
- e) Compare and contrast various modern database models, such as NoSQL and NewSQL, for handling semi-structured and unstructured data.
- f) Develop real-world database applications using advanced DBMS tools, demonstrating practical skills in performance tuning, transaction management, and data integration.

**Course Contents**

<b>Unit 1</b>	<b>Advanced SQL &amp; Transaction Management</b>	<b>6 Hours</b>
---------------	--	----------------

**Transaction Management** - Transactional properties: Atomicity, Consistency, Isolation, Durability (ACID), Transaction states and state transitions, Schedules and serializability, Concurrency control mechanisms.

**Concurrency Control** -Lock-based protocols: Two-Phase Locking (2PL), Timestamp-based protocols, Mult version Concurrency Control (MVCC), Deadlock detection and resolution. **Recovery Management** - Log-based recovery techniques, Checkpointing and crash recovery, Shadow paging, ARIES (Algorithms for Recovery and Isolation Exploiting Semantics) model, **Advanced SQL Features**- Triggers, stored procedures, and functions, Recursive queries and common table expressions (CTEs), Window functions and analytic queries, Dynamic SQL and prepared statements.

<b>Case Study#1:</b>	Transaction Management in an Online Banking System
----------------------	--

<b>Case Study#2:</b>	Advanced SQL for HR Management System
----------------------	---------------------------------------

<b>Unit 2</b>	<b>Relational Database Design &amp; Query Optimization</b>	<b>6 Hours</b>
---------------	--	----------------

**Relational Database Design**- Functional dependencies and normalization, Boyce-Codd Normal Form (BCNF), Fourth (4NF) and Fifth (5NF) Normal Forms, Decomposition and synthesis approaches **Query Processing** -Query parsing and translation, Query optimization techniques, Cost estimation and statistics, Join algorithms: Nested loop, hash join, merge join **Indexing Techniques**- B+ Tree and B-Tree indexes, Bitmap and hash indexes, Indexing for multi-dimensional data (R-trees, Quad-trees) Index maintenance and performance tuning **Query Execution Plans** - Understanding and interpreting execution plans, Cost-based optimization strategies Index selection and usage, Query rewriting and transformation.

<b>Case Study#3:</b>	Normalization and Optimization in a College Database	
<b>Case Study#4:</b>	E-Commerce Product Search Optimization	
<b>Unit 3</b>	<b>Distributed &amp; Parallel Databases</b>	<b>6 Hours</b>
<p><b>Distributed Database Systems</b>- Architectures: Client-server, peer-to-peer, hybrid, Data distribution strategies: Fragmentation, replication, allocation, Distributed query processing and optimization, Concurrency control in distributed systems, <b>Parallel Database Systems</b> - Parallel query execution models, Data partitioning and load balancing, Parallel algorithms for relational operations, Performance evaluation and scalability <b>Fault Tolerance and Recovery</b>- Failure classification and detection Checkpointing and logging in distributed systems, Recovery protocols and algorithms</p> <p>Replication and consistency models, <b>Emerging Trends in Distributed Databases</b> -</p> <p>Cloud-based distributed databases, Blockchain and decentralized databases, Edge computing and data locality, Data privacy and security considerations</p>		
<b>Case Study#5:</b>	Distributed Retail Inventory Management System	
<b>Case Study#6:</b>	Parallel Query Execution in a Cloud-Based Social Media Platform	
<b>Unit 4</b>	<b>NoSQL Databases</b>	<b>6 Hours</b>
<p><b>Introduction to NoSQL</b>- CAP Theorem: Consistency, Availability, Partition Tolerance</p> <p>BASE properties: Basically Available, Soft state, eventually consistent, Polyglot persistence and schema flexibility, Comparison with relational databases, <b>Key-Value Stores</b>- Data model and architecture, Use cases and limitations, Examples: Redis, Riak</p> <p>Design patterns and best practices, <b>Document-Oriented Databases</b>- Data model and structure, CRUD operations and querying, Indexing and aggregation frameworks Examples: MongoDB, CouchDB, <b>Column-Family Stores</b>- Data model and storage architecture, Column families and row keys, Use cases and scalability, Examples: Apache Cassandra, HBase, <b>Graph Databases</b>- Graph data model: Nodes, edges, properties, Graph traversal algorithms, Use cases in social networks, recommendation systems</p>		

<b>Case Study#7:</b>	Real-Time Analytics with Key-Value and Column-Family Stores for an Online Gaming Platform	
<b>Case Study#8:</b>	Document and Graph Database Integration for a Social Networking App	
<b>Unit 5</b>	<b>Advanced Database Architectures</b>	<b>6 Hours</b>
<b>Object-Oriented Databases:</b> Object-Relational Mapping (ORM), Object Query Language (OQL), <b>XML Databases:</b> XML Schema, XQuery, XPath, <b>Data Warehousing:</b> ETL processes, Star and Snowflake schemas, <b>Data Mining:</b> Association rules, Classification, Clustering, <b>Business Intelligence:</b> OLAP, OLTP systems		
<b>Case Study#9:</b>	Building a Business Intelligence System for a Retail Chain	
<b>Case Study#10:</b>	XML and Object-Oriented Database Integration for a Digital Library	
<b>Unit 6</b>	<b>Emerging Trends &amp; Applications</b>	<b>6 Hours</b>
<b>Cloud Databases:</b> Amazon RDS, Google Cloud SQL, Azure SQL Database, <b>Big Data Technologies:</b> Hadoop, Apache Spark, NoSQL integration, <b>Blockchain Databases:</b> Decentralized data storage, Smart contracts, <b>Graph Databases:</b> Neo4j, Cypher query language, <b>Future Trends:</b> AI in databases, Autonomous databases		
<b>Case Study#11:</b>	Cloud-Based Scalable Database for a Food Delivery Platform	
<b>Case Study#12:</b>	Blockchain and AI-Driven Healthcare Record Management System	
		<b>Total Hours:36</b>
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke</li> <li>2. "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. "Fundamentals of Database Systems" by Ramez Elmasri and Shamkant B. Navathe</li> <li>2. "Principles of Distributed Database Systems" by M. Tamer Özsu and Patrick Valduriez</li> <li>3. "Data Warehousing, Data Mining, and OLAP" by Alex Berson and Stephen J. Smith</li> </ol>		
<b>E-Books/Online Learning Resources:</b>		
<ul style="list-style-type: none"> <li>▪ <i>Advanced Database Systems</i>Instructor by Prof. S. Sudarshan (IIT Bombay)</li> <li>▪ <a href="https://www.w3schools.com/c/">https://www.w3schools.com/c/</a></li> <li>▪ <a href="https://www.programiz.com/c-programming">https://www.programiz.com/c-programming</a></li> </ul>		
<b>Online Courses:</b>		

- **Advanced Database Systems**
  - a. Instructor: Prof. S. Sudarshan (IIT Bombay)
  - b. Covers query processing, indexing, transaction management, distributed databases, NoSQL.
  - c. Link: [https://onlinecourses.nptel.ac.in/noc21\\_cs48/preview](https://onlinecourses.nptel.ac.in/noc21_cs48/preview)
- **Databases**
  - a. Instructor: Prof. Jennifer Widom (Stanford University)
  - b. Covers relational algebra, SQL, XML, NoSQL, distributed databases, transactions.
  - c. Link: <https://online.stanford.edu/courses/sohs-ydba0001-databases>
- **Advanced Databases and SQL for Data Science**
  - a. Instructor: IBM team
  - b. Covers advanced SQL, performance tuning, data modeling, transactions.
  - c. Link: <https://www.coursera.org/learn/advanced-data-science-sql>
- **Advanced SQL for Data Scientists and Analysts**
  - a. Instructor: Multiple instructors on Udemy
  - b. Covers advanced joins, indexes, query optimization, stored procedures, views.
  - c. Link: <https://www.udemy.com/course/advanced-sql-mysql>

The CO-PO Mapping Matrix

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	1	2	3	-	-	-	1	-	2
CO2	3	2	1	2	3	-	-	-	-	-	1
CO3	3	3	3	2	3	2	-	1	1	1	2
CO4	3	3	2	3	3	-	-	-	-	-	1
CO5	2	3	2	2	3	2	-	-	-	-	2
CO6	3	2	3	2	3	2	1	2	2	2	2



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Research Methodology & IPR			<b>Code</b>	DP04020125103			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEE</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	--	90	--	60		150

**Prerequisite:**

- 1) Basics of Computers
- 2) Mathematical and Statistical Foundation

**Other Courses in this Series**

NIL

**Course Objectives:**

- a) To introduce students to principles and practices of research methodology.
- b) To develop skills in formulating research problems, literature review, data analysis, and technical writing.
- c) To sensitize students to issues of ethics, plagiarism, and intellectual property rights.
- d) To foster awareness about patents, copyrights, trademarks, and the importance of innovation.
- e) To develop the ability to assess patentability and understand the patent filing process.

**Course Outcomes:**

**On completion of the course, students will be able to:**

- CO1: Understand** the fundamental principles and types of research and apply them to define research problems.
- CO2: Conduct** a structured literature review using research databases and reference management tools.
- CO3: Design** research methodology and analyse data using basic statistical tools and techniques.
- CO4: Apply** ethical principles in research and demonstrate skills in writing research papers, proposals, and reports.

**CO5: Explain** various forms of Intellectual Property Rights and understand the process of patent filing in the Indian and international context.

**CO6: Explore** the role of innovation and entrepreneurship in converting research outcomes into real-world applications.

Course Contents		
<b>Unit 1</b>	<b>Introduction to Research Methodology</b>	<b>6 Hours</b>
<p><b>Introduction</b> : concept, nature, and objectives of research, focusing on different types such as fundamental, applied, qualitative, and quantitative research. It covers the entire research process, including problem identification, formulation of research questions, and criteria for good research. Emphasis is placed on the role and scope of research in computer science and engineering with relevant examples.</p>		
<b>Case Study#1:</b>	<p><b><i>Problem Formulation in Machine Learning</i></b></p> <p>A postgraduate student explored improving the accuracy of spam email detection using NLP. Initial problem identification involved classifying emails using existing models. Through refinement, the student framed the research problem as “enhancing accuracy of spam detection in multilingual datasets using deep learning techniques.”</p>	
<b>Case Study#2:</b>	<p><b><i>Defining a Research Problem in Cybersecurity</i></b></p> <p>A researcher identifies a surge in ransomware attacks targeting educational institutions. The problem is scoped into a research question: “What are the most effective real-time detection strategies for ransomware in campus networks?”</p>	
<b>Unit 2</b>	<b>Research Design &amp; Literature Review</b>	<b>6 Hours</b>
<p>Research Design and its Types, Sources of Literature: Journals, Conferences, Digital Libraries ,Literature Review Techniques,Referencing Styles (APA, IEEE, etc.),Tools for Managing References (Zotero, Mendeley), Gap Analysis and Problem Definition</p>		
<b>Case Study#3:</b>	<p><b><i>Identifying a Gap in Fake News Detection Research</i></b></p> <p>While reviewing literature on fake news detection, a student discovers limited work in regional Indian languages. This gap helps define a research objective: "Develop a multilingual model for fake news detection."</p>	

<p><b>Case Study#4:</b></p>	<p><b>Research Design in Human-Computer Interaction [HCI]</b></p> <p>While studying usability of mobile apps for elderly users, the student used descriptive research design and created a user group to test accessibility features, aligning methodology with research objectives.</p>	
<p><b>Unit 3</b></p>	<p><b>Data Collection and Analysis</b></p>	<p><b>6 Hours</b></p>
<p>Data Types and Sources: Primary, Secondary, Qualitative and Quantitative Data Collection, Surveys, Interviews, Questionnaires, Sampling Techniques, Data Processing and Visualization, Introduction to Statistical Tools (SPSS, R, Excel)</p>		

<p><b>Case Study#5:</b></p>	<p><b>Sentiment Analysis of Social Media</b></p> <p>A research team collected Twitter data using APIs, applied sentiment classification techniques, and analyzed data using Excel and R. The project revealed shifting public sentiment trends on climate change awareness.</p>	
<p><b>Case Study#6:</b></p>	<p><b>Data Analysis in Traffic Prediction Using IoT</b></p> <p>Real-time traffic data is collected using sensors at a busy intersection. The student analyzes this data using Python to predict peak traffic hours and model congestion patterns using regression analysis.</p>	
<p><b>Unit 4</b></p>	<p><b>Research Ethics and Technical Writing</b></p>	<p><b>6 Hours</b></p>
<p>Ethics in Research: Integrity, Confidentiality, Consent, Plagiarism: Forms, Detection Tools (Turnitin, Grammarly), UGC Guidelines and Ethics Committees, Structure of a Research Paper/Thesis, Writing Proposals, Abstracts, and Reports, Effective Presentation Techniques</p>		
<p><b>Case Study#7:</b></p>	<p><b>Plagiarism Detection in Thesis Writing</b></p> <p>A postgraduate student submits a research paper with copied content from online blogs. Using Turnitin, the guide identifies 42% plagiarism. The student was guided to revise with proper paraphrasing and references. The student rewrites and learns proper citation techniques.</p>	

<p><b>Case Study#8</b></p>	<p><b><i>Ethical Dilemma in Biomedical Research</i></b></p> <p>In a study involving emotion detection via facial recognition, ethical concerns arise around data privacy and consent. The institution’s review board intervenes to ensure ethical practices in data collection.</p>	
<p><b>Unit 5</b></p>	<p><b>Introduction to IPR</b></p>	<p><b>6 Hours</b></p>
<p>Definition and Importance of IPR, Types of IPR Patents, Copyrights, Trademarks, Trade Secrets ,Indian and International IPR Frameworks WIPO, TRIPS, Filing Procedure for Patents in India ,Case Studies: Patents in IT and Computer Science, Open-Source Licensing and Software Patents.</p>		
<p><b>Case Study#9:</b></p>	<p><b><i>Patent Filing for a latest Technology Product</i></b></p> <p>An M.Tech student invents a gesture-controlled home automation system and files a patent through the Indian Patent Office with the help of the college IPR cell.</p>	
<p><b>Case Study#10:</b></p>	<p><b><i>Copyright Violation in Software Reuse</i></b></p> <p>A startup modifies open-source code without adhering to the GPL license terms. Legal notice prompts a review of software licensing, and the startup switches to a proper attribution model.</p>	
<p><b>Unit 6</b></p>	<p><b>Innovation and Entrepreneurship in Research</b></p>	<p><b>6 Hours</b></p>

<p>Role of Innovation in Research, Technology Transfer, Startups, and Incubation, Government Schemes and Support Startup India, MSME, Commercialization of Research, IPR and Funding Proposals.</p>		
<p><b>Case Study#11:</b></p>	<p><b><i>Prepare a Presentation and Document a Real-Time Startup Success Story</i></b></p>	
<p style="text-align: right;"><b>Total Hours: 36</b></p>		
<p><b>Text Books:</b></p>		
<ol style="list-style-type: none"> <li>1. Kothari, C. R., &amp; Garg, G. 2019, “Research methodology: Methods and techniques”,4th edition , New Age International Publishers , ISBN 9789386649225</li> <li>2. Kumar, R. 2019 ,“Research methodology: A step-by-step guide for beginners”, 5th edition SAGE Publications, ISBN 1526449897</li> <li>3. Soumitro Banerjee, “Research Methodology for Natural Sciences”, IISc Press, ISBN 8194835178</li> </ol>		

<b>Reference Books:</b>											
<p>a) Gopalakrishnan, N. S., &amp; Agitha, T. G. [2009]. Principles of intellectual property, 2nd Edition Eastern Book Company , ISBN [978-9351450191</p> <p>b) Basant, K., &amp; Venugopal, K. N. [2013]. Patent law and practice, LexisNexis Butterworths.</p>											
<b>E-Books/Online Learning Resources:</b>											
<p>a) Byers, T., Dorf, R. C., &amp; Nelson, A. [2020]. <i>Technology Ventures: From Idea to Enterprise</i> [5th ed.]. McGraw-Hill Education. <a href="https://www.mheducation.com/highered/product/technology-ventures-dorf-byers/M9781259861424.html">https://www.mheducation.com/highered/product/technology-ventures-dorf-byers/M9781259861424.html</a></p> <p>b) Poltorak, A. I., &amp; Lerner, P. J. [2011]. <i>Essentials of Intellectual Property: Law, Economics, and Strategy</i>. WIPO. Available online via WIPO Knowledge Repository <a href="https://tind.wipo.int/record/27820?utm_source=chatgpt.com">https://tind.wipo.int/record/27820?utm_source=chatgpt.com</a></p> <p>c) World Intellectual Property Organization. [2004]. WIPO intellectual property handbook: Policy, law and use. WIPO. <a href="https://www.wipo.int/publications/en/details.jsp?id=275">https://www.wipo.int/publications/en/details.jsp?id=275</a></p>											
<b>Online Courses:</b>											
<p>NPTEL [Research Methodology</p> <p>a) Instructor: Prof. Soumitro Banerjee [IISER Kolkata)</p> <p>b) The course covers all the conceptual and methodological issues that go into successful conduction of research.</p> <p>c) Link: <a href="https://onlinecourses.nptel.ac.in/noc22_ge08/preview">https://onlinecourses.nptel.ac.in/noc22_ge08/preview</a></p> <p>Udemy- Research Methodology</p> <p>a) Instructor: Sam Mahmud</p> <p>b) Research Problem, Research Design, Sampling Techniques, Research Proposal, Data Collection, Data Analysis, and Research Report</p> <p>c) Link: <a href="https://www.udemy.com/share/101w7c/">https://www.udemy.com/share/101w7c/</a></p>											
<b>The CO PO Mapping Matrix</b>											
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	3	-	2	-	-	-	-	-	1	2
CO2	2	3	-	3	2	-	-	-	-	1	2
CO3	2	3	2	3	2	-	-	-	-	-	2
CO4	-	-	-	2	-	2	-	3	2	3	2
CO5	-	-	-	-	-	2	2	3	-	2	2
CO6	-	2	2	2	3	2	2	-	2	2	3



<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Advances in Algorithms Lab			<b>Code</b>	DP04020125104			
	Teaching Scheme (Hrs/Week)			Examination Scheme & Marks				
<b>Credits</b>	Lectures	Tutorials	Practical	CA		TEA		Total
				Theory	Practical	Theory	Practical	
1	--	--	2	--	--	--	50	50
<b>Prerequisite:</b>								
<ul style="list-style-type: none"> <li>d. Discrete Mathematics</li> <li>e. Data Structure and Algorithms</li> <li>f. Theory of Computation</li> <li>g. Design and Analysis of Algorithms</li> </ul>								
<b>Other Courses in this Series:</b>								
Advances in Algorithms								
<b>Course Objectives:</b>								
<ol style="list-style-type: none"> <li>1. To understand performance of algorithms and its correctness.</li> <li>2. To apply various algorithmic strategies in solving computational problems effectively.</li> <li>3. To evaluate and compare the performance of algorithmic approaches based on time and space complexity.</li> <li>4. To design algorithms that are optimized for both time and space efficiency.</li> <li>5. To understand the principles and implementation of parallel and distributed algorithms.</li> </ol>								
<b>Course Outcomes:</b>								
On completion of the course, students will be able to:								
<b>CO1:</b> Analyse performance of algorithms.								
<b>CO2:</b> Identify suitable algorithmic strategies and Apply them effectively to solve the given problem								

**CO3:** Apply Algorithmic Strategies to solve complex computational problems and Evaluate their efficiency and correctness.

**CO4:** Implement different strategies to solve the problem optimally.

**CO5:** Understand and Analyze graph flow and matching algorithms to solve real time problems.

**CO6:** Understand and apply algorithmic solutions to real-life problems with Implementation and optimization in distributed and parallel environments.

Suggested List of Laboratory Experiments/Assignments

Assignments from all Groups ( A, B, C, D, E, and F) are compulsory

Programming Tools Recommended: C++/ Python / Java /MATLAB/ OpenMP/MPI

<b>Sr. No.</b>	<b>Group-A (Unit-1 Algorithms and Complexity Theory)</b>
1)	<p><b>Solve Any One:</b></p> <p>A. Implement Merge Sort, Quick Sort, and Bubble Sort. Analyse their performance for input sizes ranging from 100 to 10,000 elements.</p> <p>B. Design and implement a program to solve the Subset Sum Problem, a classical NP-complete problem. Consider a set of integers and a target sum.</p> <p>C. Given an undirected graph, find a vertex cover using approximation algorithm.</p>
<b>Sr. No.</b>	<b>Group B (Unit-2 Greedy and Dynamic Programming Strategy)</b>
2)	<p><b>Solve Any One:</b></p> <p>A. Implement Prim's Algorithm and Kruskal's Algorithms for finding Minimum Spanning Tree.</p> <p>B. Implement Matrix Chain Multiplication using dynamic programming.</p> <p>C. Write a program to solve a 0-1 Knapsack problem using dynamic programming</p>
<b>Sr. No.</b>	<b>Group-C (Unit-3 Algorithmic Strategies)</b>
3)	<p><b>Solve Any One:</b></p> <p>A. Implement Randomized Quick Sort and compare its performance with deterministic Quick Sort.</p> <p>B. Implement an algorithm to colour a given graph using backtracking such that no two adjacent vertices have the same colour</p> <p>C. Implement both Monte Carlo (e.g., primality test) and Las Vegas (e.g., randomized selection) algorithms. Compare their accuracy, performance, and behaviour.</p>
<b>Sr. No.</b>	<b>Group-D (Unit-4 Abstract Algorithms)</b>
4)	<p><b>Solve Any One:</b></p>

	<p>A. Implement a Genetic Algorithm to find the global maximum of the function <math>f(x) = x * \sin(10\pi x) + 1.0</math> for <math>x \in [0, 1]</math>. Consider selection, crossover, mutation.</p> <p>B. Implement exponential temperature decay and probabilistic neighbour acceptance(TSP) using Simulated Annealing.</p> <p>C. Implement Ant Colony Optimization to find the shortest path between nodes in a weighted undirected graph with 10 nodes.</p>
<b>Sr. No.</b>	<b>Group-E (Unit-5 Graph Flows and Matching Algorithms)</b>
5)	<p><b>Solve Any One:</b></p> <p>A. Implement the Ford-Fulkerson algorithm to compute the maximum flow in a directed flow network.</p> <p>B. Implement the Naïve String-Matching algorithm and Finite Automata-Based Matching to find all occurrences of a pattern P in a text T. Consider any string as a input.</p> <p>C. Implement edit distance calculation (Levenshtein distance) using dynamic programming to allow approximate string matches.</p>
<b>Sr. No.</b>	<b>Group-F (Unit-6 Parallel and Distributed Algorithms)</b>
6)	<p><b>Solve Any One:</b></p> <p>A. Implement Parallel Merge Sort using multi-threading and multiprocessing.</p> <p>B. Design and implement Bitonic Sort in parallel. Use suitable data/inputs</p> <p>C. Simulate Distributed Depth First Search using message passing (synchronous or asynchronous).</p>
<b>Sr. No.</b>	<b>Mini Projects: Design and Develop any one Mini-Project from Sr. No. 1 to 2</b>
1)	Develop a system to perform parallel PageRank or Community Detection on large-scale graphs using MPI or OpenMP.
2)	Design a Parallel Genetic Algorithm to optimize job scheduling in a simulated cloud environment.
<b>Text Books:</b>	
Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", MIT Press; ISBN 978-0-262-03384-8.	
<b>Reference Books:</b>	
<ol style="list-style-type: none"> <li>Horowitz and Sahani, "Fundamentals of Computer Algorithms", University Press, ISBN: 978 817371 6126, 81 7371 61262.</li> <li>Aho, Hopcroft, Ullman "The Design and Analysis of Computer Algorithms".</li> <li>G. Brassard, P. Bratley, "Fundamentals of Algorithmics", PHI</li> </ol>	

**E-Books/Online Learning Resources:**

1. [https://homel.vsb.cz/~fai0013/Kniha\\_Algoritmy.pdf](https://homel.vsb.cz/~fai0013/Kniha_Algoritmy.pdf)
2. <https://www.phindia.com/Books/BookDetail/9788120352124/parallel-and-distributed-computing-basu?srsId=AfmBOopWwGRLJnpIVcAEoi65RkYe4fcQBYD6CRDJcFtCl7IQwIHTHWf9>

**Online Courses:**

- 1. Design and Analysis of Algorithm**
  - d. Instructor : Madhavan Mukund
  - e. Cover : Design and Analysis of Algorithm
  - f. Link : <https://nptel.ac.in/courses/106106131>
- 2. Design and Analysis of Algorithm**
  - d. Instructor : Bhanu Priya
  - e. Cover : Algorithm Introduction
  - f. Link : <https://youtu.be/GQNT0v5zKhE?si=IAvx3r3me1EQZrSD>
- 3. Programming, Data Structures and Algorithms**
  - d. Instructor : N. S. Narayanaswamy
  - e. Cover : Programming, Data Structures and Algorithms
  - f. Link : <https://youtu.be/IW-1-TQZsMw?si=SFo-gmTTQiLUBbOs>
- 4. Graph Searching, BFS and DFS**
  - h. Instuctor : Prof. Benny George K and Prof. Sajith Gopalan
  - i. Cover : Graph Searching, BFS and DFS  
Link : <https://youtu.be/u2moDPvDmac?si=b3ffPaX9fh4wMzMd>

**Software Tools, Simulation Platforms & Hardware Kits:**

1. GCC (GNU Compiler Collection) – Standard C compiler for Linux
2. Code::Block Open-source C/C++ IDE with built-in compiler support. <https://www.codeblocks.org/>
3. OnlineGDB (<https://www.onlinegdb.com/>) – Online compiler and debugger.
4. CodeChef IDE (<https://www.codechef.com/ide>) – Competitive coding and practice platform.

**The CO-PO Mapping Matrix**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	3	2	-	-	-	-	-	2	-
CO2	3	3	3	2	2	-	-	-	-	-	2
CO3	2	3	3	2	3	-	-	-	-	-	2
CO4	2	2	3	3	3	-	-	-	-	-	2
CO5	3	3	3	3	2	3	-	-	-	-	-
CO6	2	2	2	2	2	3	-	-	-	-	3



<b>Program</b>	M. Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Advanced Database Management System Lab			<b>Code</b>	DP04020125105			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
1	--	--	2	--	--	--	50	50
<b>Prerequisite:</b>								
1) Database Management Systems								
<b>Other Courses in this Series:</b>								
1. Database Management Systems Laboratory								
<b>Course Objectives:</b>								
<p><b>CO1:</b> To provide a deep understanding of advanced database concepts such as query optimization, indexing techniques, and transaction management for efficient data handling.</p> <p><b>CO2:</b> To explore the architecture and functioning of distributed databases, enabling students to design systems that support data distribution and replication across multiple nodes.</p> <p><b>CO3:</b> To introduce emerging database technologies, including NoSQL, NewSQL, and columnar databases, for managing semi-structured and unstructured data.</p> <p><b>CO4:</b> To study the techniques of query processing and optimization, helping students understand how complex SQL queries are parsed, interpreted, and optimized by the database engine.</p> <p><b>CO5:</b> To develop the ability to analyze and implement concurrency control and recovery mechanisms, ensuring database consistency and reliability in multi-user environments.</p> <p><b>CO6:</b> To enable students to work with real-world data systems, including hands-on experience with advanced DBMS tools, big data platforms, and modern database APIs.</p>								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<ol style="list-style-type: none"> <li>1. Analyse and optimize complex SQL queries using advanced query processing techniques and cost-based optimization strategies</li> <li>2. Apply appropriate indexing and hashing techniques to improve database query performance and retrieval speed.</li> <li>3. Design and implement distributed database systems with concepts of data fragmentation, replication, and</li> </ol>								

distributed query processing.

4. Evaluate concurrency control mechanisms and recovery strategies to ensure database consistency and fault tolerance in multi-user environments.
5. Compare and contrast various modern database models, such as NoSQL and NewSQL, for handling semi-structured and unstructured data.
6. Develop real-world database applications using advanced DBMS tools, demonstrating practical skills in performance tuning, transaction management, and data integration.

#### **Suggested List of Laboratory Experiments/Assignments**

**Assignments from all Groups ( A, B, C, D, E, and F) are compulsory**

Sr. No.	Group-A (UNIT 1 Advanced SQL & Transaction Management)
	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1. Create and execute SQL transactions on a sample database to illustrate the four ACID properties—Atomicity, Consistency, Isolation, and Durability—by simulating real-world scenarios such as money transfers or inventory updates.</li> <li>2. Use SQL or PL/SQL to create a simulated transaction that moves through all key transaction states: begin with an active state, perform operations to reach partially committed, then either commit for success or trigger an error to simulate failure and rollback to aborted.</li> <li>1. Implement and analyze different transaction schedules in SQL to compare for conflict serializability and view serializability, and determine whether each schedule is serializable.</li> <li>2. Simulate the Two-Phase Locking (2PL) protocol by executing multiple transactions on shared data, ensuring proper lock acquisition and release phases to maintain serializability and avoid conflicts.</li> <li>3. Demonstrate a timestamp-based concurrency control protocol by assigning timestamps to transactions and executing them based on ordering rules to maintain consistency and prevent conflicts.</li> <li>4. Use a database system that supports MVCC (such as PostgreSQL) to run concurrent transactions and observe how the system handles multiple versions of data for consistent and isolated reads and writes.</li> <li>5. Simulate a deadlock scenario by creating interdependent transactions and implement deadlock detection techniques such as wait-for graphs; then apply resolution strategies like transaction rollback or timeout.</li> </ol>
	<ol style="list-style-type: none"> <li>6 Implement Write-Ahead Logging (WAL) in a simulated environment and demonstrate recovery of committed transactions after a system failure."</li> <li>7 Simulate the creation of checkpoints during transaction processing and demonstrate how the system can recover to a consistent state after a simulated crash.</li> <li>8 Create a file-based storage simulation to demonstrate how shadow paging ensures atomicity</li> </ol>

	<p>and durability without the need for a log.</p> <p>9 Simulate the key phases of the ARIES recovery algorithm—Analysis, Redo, and Undo—using example transaction logs to recover a database to a consistent state after failure.</p> <p>10 Create BEFORE and AFTER triggers in a database to automate actions such as logging changes or validating data during INSERT, UPDATE, or DELETE operations.</p> <p>11 Write and execute stored procedures and functions in SQL/PLSQL to encapsulate and reuse business logic within the database.</p> <p>12 Use Common Table Expressions (CTEs), including recursive CTEs, to query hierarchical or tree-structured data such as an organizational chart or category-subcategory relationships.</p> <p>13 Use SQL window functions such as RANK(), DENSE_RANK(), ROW_NUMBER(), LEAD(), and LAG() to perform advanced analytics over partitions of data.</p> <p>14 Implement dynamic SQL to allow flexible query construction at runtime and demonstrate the use of prepared statements to improve performance and prevent SQL injection attacks.</p>
<b>Sr. No.</b>	<b>Group-B (Unit 2 Relational Database Design &amp; Query Optimization)</b>
<b>1.</b>	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>Analyze a sample dataset to identify and document all functional dependencies between attributes.</li> <li>Perform step-by-step normalization of a given relational schema from First Normal Form (1NF) through to Boyce-Codd Normal Form (BCNF), showing each stage clearly.</li> <li>Extend normalization of a relational schema to Fourth Normal Form (4NF) and Fifth Normal Form (5NF) by addressing multivalued dependencies and join dependencies.</li> <li>Apply decomposition techniques on a relational schema to achieve lossless and dependency-preserving decompositions, and use synthesis methods to reconstruct normalized schemas.</li> <li>Manually convert SQL queries into equivalent relational algebra expressions and intermediate representations used in query processing.</li> <li>Demonstrate the logical and physical query optimization process by tracing query execution plans using database tools or manual step-by-step analysis.</li> <li>Estimate the cost of different query execution plans by analyzing table statistics such as cardinality and selectivity to determine the most efficient plan.</li> <li>Implement or simulate different join algorithms including Nested Loop Join, Hash Join, and Merge Join to compare their performance and use cases.</li> </ol>
<b>2.</b>	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>Create B-Tree and B+ Tree indexes on sample tables and measure their impact on query performance for both point queries and range queries.</li> <li>Implement bitmap indexes on low-cardinality columns and hash indexes for equality-based</li> </ol>

	<p>searches, then evaluate their effectiveness.</p> <ol style="list-style-type: none"> <li>3) Explore or simulate multi-dimensional indexing techniques such as R-tree and Quad-tree for efficient spatial query processing.</li> <li>4) Perform INSERT, UPDATE, and DELETE operations on indexed tables and analyze how these operations affect index structure and query performance.</li> <li>5) Use SQL tools like EXPLAIN or EXPLAIN ANALYZE to generate and interpret execution plans for different queries.</li> <li>6) Compare and analyze execution plans for alternative query strategies, focusing on estimated costs and resource usage.</li> <li>7) Analyze how different index choices affect the performance of real or simulated queries by monitoring execution times and plan changes.</li> <li>8) Rewrite SQL queries—such as converting subqueries to joins or rearranging join orders—to improve performance without altering query results.</li> </ol>
<b>Sr. No.</b>	<b>Group-C (Unit 3 Distributed &amp; Parallel Databases)</b>
<b>1.</b>	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1. Set up and demonstrate distributed systems using client-server, peer-to-peer, and hybrid architectures.</li> <li>2. Implement and test data distribution methods by applying horizontal and vertical fragmentation, full and partial replication, and distributing data fragments across different locations.</li> <li>3. Write and run distributed database queries that minimize the amount of data transferred between sites.</li> <li>4. Compare and evaluate different distributed query plans to choose the most efficient and cost-effective solution.</li> <li>5. Simulate concurrency problems in distributed systems and apply solutions like distributed two-phase locking, timestamp ordering, and deadlock detection and resolution to manage them.</li> </ol>
<b>2.</b>	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1) Simulate or analyze query execution using pipelined and partitioned parallelism to understand different parallel processing models.</li> <li>2) Implement horizontal data partitioning and demonstrate how to distribute workloads evenly to achieve effective load balancing.</li> <li>3) Develop and test parallel algorithms for relational database operations such as joins, selections, and aggregations.</li> <li>4) Measure and evaluate the performance of parallel query executions by analyzing speed-up, scale-up, and size-up on simulated or cloud-based parallel systems.</li> <li>5) Simulate different types of system failures—including crash, omission, and Byzantine faults—and use health-check tools to detect these failures.</li> </ol>

	6) Demonstrate checkpointing techniques by creating and recovering checkpoints in a distributed system to ensure fault tolerance.
3.	<p><b>Solve any one</b></p> <ol style="list-style-type: none"> <li>1. Implement or simulate logging and recovery protocols like ARIES or other distributed recovery algorithms using transaction logs.</li> <li>2. Simulate replication and consistency models such as eventual consistency, strong consistency, and quorum-based replication to observe their behavior.</li> <li>3. Use cloud database services like Google Firestore or Amazon Aurora to set up distributed databases that demonstrate data distribution and scalability.</li> <li>4. Build and deploy a basic blockchain system to implement a decentralized ledger for distributed data storage.</li> <li>5. Simulate edge computing scenarios where edge nodes process local data and synchronize updates with a central system.</li> <li>6. Apply data privacy and security measures by implementing encryption, access control, and auditing features in a distributed database environment to ensure compliance.</li> </ol>
<b>Sr. No.</b>	<b>Group-D (Unit 4: NoSQL Databases)</b>
1.	<p><b>Solve any one</b></p> <ol style="list-style-type: none"> <li>1) Simulate the trade-offs between consistency, availability, and partition tolerance (CAP theorem) using distributed database tools.</li> <li>2) Set up practical demonstrations to compare ACID properties with BASE, highlighting eventual consistency in distributed systems.</li> <li>3) Build a small application that integrates multiple databases (e.g., Redis for caching and MongoDB for document storage) to showcase polyglot persistence.</li> <li>4) Perform CRUD operations on both SQL and NoSQL databases, then compare their flexibility, performance, and schema design differences.</li> <li>5) Set up a key-value store and explore its data storage principles and architecture.</li> <li>6) Execute CRUD operations in Redis, including inserting, updating, deleting, and reading data.</li> <li>7) Implement a basic caching or session management system using Redis to demonstrate real-world use cases.</li> <li>8) Use Redis to implement design patterns such as pub/sub messaging, rate limiting, or leaderboards.</li> <li>9) Design a nested document schema in MongoDB for an e-commerce application to practice data modeling.</li> <li>10) Practice CRUD operations and querying on MongoDB documents to reinforce understanding of NoSQL data manipulation.</li> </ol>

2.	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1. Experiment with schema-less design by storing different document structures within the same collection to understand schema flexibility.</li> <li>2. Set up a column-family database, define column families and row keys, and explore the column-family data model.</li> <li>3. Create indexes and perform complex queries using aggregation pipelines to optimize query performance.</li> <li>4. Perform CRUD operations using Cassandra Query Language (CQL) in Apache Cassandra.</li> <li>5. Demonstrate data distribution, replication, and scalability features in a column-family store.</li> <li>6. Model graph data by creating nodes and relationships for a social network dataset.</li> <li>7. Execute graph queries using Cypher in Neo4j to perform CRUD and more complex graph operations.</li> <li>8. Implement graph traversal algorithms like Breadth-First Search (BFS), Depth-First Search (DFS), or pathfinding queries.</li> <li>9. Build and query applications such as recommendation systems or network analysis tools using graph databases.</li> </ol>
<b>Sr. No.</b>	Group-E (Unit 5: Advanced Database Architectures )
1)	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>10. Use an ORM tool like Hibernate or SQLAlchemy to map class objects to relational database tables and perform CRUD operations with object-oriented code.</li> <li>11. Run OQL-like queries by using an object database or simulating them through ORM query methods.</li> <li>12. Design XML schemas (XSD) and validate XML documents to ensure they follow the defined structure.</li> <li>13. Write and execute XPath expressions to extract specific data from XML documents.</li> <li>14. Use XQuery to perform advanced queries and transform XML datasets.</li> <li>15. Build and run a basic ETL pipeline using tools like Talend, Pentaho, or Python libraries such as Pandas.</li> </ol>
2)	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1. Design a star schema for a retail or sales dataset to organize data for analysis.</li> <li>2. Create a snowflake schema, then compare its structure and benefits with the star schema.</li> <li>3. Apply association rule mining algorithms like Apriori or FP-Growth using tools such as Weka, Orange, or Python's mlxtend library.</li> <li>4. Use classification algorithms such as decision trees, k-NN, or SVM to classify sample datasets.</li> <li>5. Perform clustering on datasets using k-means or hierarchical clustering methods.</li> <li>6. Use OLAP tools (e.g., Microsoft SSAS or Pentaho) to perform operations like slice, dice, roll-up, and drill-down on OLAP cubes.</li> <li>7. Build a basic OLTP system, such as a point-of-sale or inventory system, supporting real-time</li> </ol>

	<p>transaction processing.</p> <p>8. Create a BI dashboard using platforms like Tableau, Power BI, or Google Data Studio to visualize business data.</p>
<b>Sr. No.</b>	<b>Group-F (Unit 6: Emerging Trends &amp; Applications)</b>
1)	<p><b>Solve any one.</b></p> <ol style="list-style-type: none"> <li>1) Launch an Amazon RDS instance, connect to it, and perform CRUD operations on the database.</li> <li>2) Deploy a Google Cloud SQL instance, set up access permissions, and execute SQL queries.</li> <li>3) Create an Azure SQL Database, import sample datasets, and run basic SQL queries.</li> <li>4) Demonstrate backup and restore processes using one of the cloud database services (e.g., AWS, Google Cloud, or Azure).</li> <li>5) Install Hadoop and run basic HDFS commands to manage distributed storage.</li> <li>6) Write and execute a MapReduce program, such as a word count application, on a Hadoop cluster.</li> <li>7) Perform data loading, transformation, and actions in Apache Spark using PySpark or Scala.</li> <li>8) Integrate a NoSQL database like MongoDB or Cassandra with Spark for big data processing tasks.</li> <li>9) Deploy a simple blockchain network using frameworks such as Hyperledger Fabric or an Ethereum testnet to explore decentralized data storage.</li> </ol>
	<p><b>Solve any one.</b></p> <ul style="list-style-type: none"> <li>) Write, deploy, and interact with a basic smart contract using Solidity on a blockchain platform like Ethereum.</li> <li>) Install and set up Neo4j, then load sample datasets to prepare for graph database operations.</li> <li>) Write Cypher queries to create nodes, define relationships, and perform graph traversals in Neo4j.</li> <li>) Implement graph analytics use cases, such as social network analysis or recommendation queries, using Neo4j.</li> <li>) Explore AI integration in databases by experimenting with AI-driven indexing or query optimization via cloud AI services or third-party tools.</li> <li>) Work hands-on with autonomous databases like Oracle Autonomous Database to experience features such as self-tuning, automatic patching, and dynamic scaling.</li> </ul>
<b>Sr. No.</b>	<b>Mini Projects: Design and Develop any one Mini-Project from Sr. No. 1 to 3</b>
1.	<ol style="list-style-type: none"> <li>1. Develop a Transaction Management and Concurrency Control system that handles simultaneous transactions, ensuring data consistency and isolation using techniques like locking protocols or timestamp ordering.</li> <li>2. Implement a Recovery Management system with Advanced SQL features, focusing on transaction logging, crash recovery, and advanced query writing for robust database operations.</li> <li>3. Design a Relational Database and optimize queries, including creating normalized database schemas and applying query optimization strategies to improve performance.</li> </ol>
<b>CO1:</b>	1. Build and simulate a Distributed Database System with Parallel Processing, focusing on data

	<p>distribution, query execution, and performance optimization across multiple nodes.</p> <ol style="list-style-type: none"> <li>2. Develop a project demonstrating NoSQL databases and Polyglot Persistence, integrating different types of databases (e.g., document store, key-value store) within a single application.</li> <li>3. Design and implement Object-Oriented and XML Databases, including schema design, querying using object query languages, and XML data manipulation.</li> </ol>
1.	<ol style="list-style-type: none"> <li>1) Design and implement a Data Warehousing and Business Intelligence system, including schema design (star or snowflake), ETL processes, and creation of BI dashboards for data analysis.</li> <li>2) Develop a Cloud Database solution integrated with Big Data technologies, demonstrating scalable storage, data processing, and real-time analytics.</li> <li>3) Build and deploy a Blockchain-based Decentralized Database, implementing smart contracts and distributed ledger functionalities for secure data management.</li> <li>4) Explore Future Trends by working on AI-powered and Autonomous Database systems, showcasing features like self-tuning, automated query optimization, and intelligent indexing.</li> </ol>
<b>Text Books:</b>	
<ol style="list-style-type: none"> <li>A. "Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke</li> <li>B. "Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan</li> </ol>	
<b>Reference Books:</b>	
<ol style="list-style-type: none"> <li>A. "Fundamentals of Database Systems" by Ramez Elmasri and Shamkant B. Navathe</li> <li>B. "Principles of Distributed Database Systems" by M. Tamer Özsu and Patrick Valduriez</li> <li>C. "Data Warehousing, Data Mining, and OLAP" by Alex Berson and Stephen J. Smith</li> </ol>	
<b>E-Books/Online Learning Resources:</b>	
<ol style="list-style-type: none"> <li>A. <i>Advanced Database Systems</i> Instructor by Prof. S. Sudarshan (IIT Bombay)</li> <li>B. <a href="https://www.w3schools.com/c/">https://www.w3schools.com/c/</a></li> <li>C. <a href="https://www.programiz.com/c-programming">https://www.programiz.com/c-programming</a></li> </ol>	
<b>Online Courses:</b>	
<ol style="list-style-type: none"> <li>A. <b>Advanced Database Systems</b> <ol style="list-style-type: none"> <li>a. Instructor: Prof. S. Sudarshan (IIT Bombay)</li> <li>b. Covers query processing, indexing, transaction management, distributed databases, NoSQL.</li> <li>c. Link: <a href="https://onlinecourses.nptel.ac.in/noc21_cs48/preview">https://onlinecourses.nptel.ac.in/noc21_cs48/preview</a></li> </ol> </li> <li>B. <b>Databases</b> <ol style="list-style-type: none"> <li>a. Instructor: Prof. Jennifer Widom (Stanford University)</li> <li>b. Covers relational algebra, SQL, XML, NoSQL, distributed databases, transactions.</li> <li>c. Link: <a href="https://online.stanford.edu/courses/sohs-ydba0001-databases">https://online.stanford.edu/courses/sohs-ydba0001-databases</a></li> </ol> </li> <li>C. <b>Advanced Databases and SQL for Data Science</b> <ol style="list-style-type: none"> <li>a. Instructor: IBM team</li> <li>b. Covers advanced SQL, performance tuning, data modeling, transactions.</li> </ol> </li> </ol>	

c. Link: <https://www.coursera.org/learn/advanced-data-science-sql>

**D. Advanced SQL for Data Scientists and Analysts**

a. Instructor: Multiple instructors on Udemy

b. Covers advanced joins, indexes, query optimization, stored procedures, views.

c. Link: <https://www.udemy.com/course/advanced-sql-mysql>

**Software Tools, Simulation Platforms & Hardware Kits:**

**Software Tools:**

- A. MySQL / PostgreSQL / Oracle DB / Microsoft SQL Server – For implementing relational DBMS.
- B. MongoDB / Cassandra / Neo4j – For NoSQL and graph-based databases.
- C. pgAdmin / DBeaver / SQL Developer – GUI tools for database design and management.
- D. Apache Hadoop / Apache Hive / Spark SQL – For large-scale data processing and querying.
- E. ER/Studio / dbForge Studio – For database modeling and design.
- F. SQLite – Lightweight embedded DB for simulations and learning.

**Simulation Platforms:**

- A. Oracle VirtualBox / Docker – To simulate database servers.
- B. Cloud Simulators: Google Cloud / AWS RDS free tier for cloud-based DBMS simulation.
- C. Sim4db / DB-Main – Academic tools for simulating and teaching DBMS concepts.

**Hardware Kits:**

Database systems are primarily software-driven, so hardware kits are not typically used. However:

- 1) Raspberry Pi / Intel NUC – Can be used to simulate distributed DBMS in a small cluster environment.
- 2) Server-grade machines or VMs – For enterprise-scale DBMS performance testing.

**The CO-PO Mapping Matrix**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	-	-	-	-	2	-	3	3
CO2	3	3	3	-	-	-	-	2	2	3	3
CO3	3	3	3	-	-	-	-	-	2	3	3
CO4	3	3	3	-	-	-	-	-	-	3	3
CO5	3	3	3	-	-	-	2	2	-	3	3
CO6	3	3	3	-	-	-	2	-	2	3	3



<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Advanced Data Science and Analytics			<b>Code</b>	DP04020125106			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	--	0	--	60	--	150
<b>Prerequisite:</b>								
<ol style="list-style-type: none"> <li>1. Fundamentals of Data Structures and Algorithms</li> <li>2. Basics of Programming (Python/R preferred)</li> <li>3. Probability and Statistics</li> <li>4. Database Management Systems</li> </ol>								
<b>Other Courses in this Series:</b>								
<ol style="list-style-type: none"> <li>1. Advanced Data Science and Analytics Lab</li> </ol>								
<b>Course Objectives:</b>								
<ol style="list-style-type: none"> <li>6. To master advanced data science and analytics techniques, including machine learning, deep learning, and real-time data processing.</li> <li>7. To design and implement robust data preprocessing, feature engineering, and visualization pipelines for complex datasets.</li> <li>8. To apply advanced supervised and unsupervised learning algorithms to solve real-world analytics problems.</li> <li>9. To leverage big data frameworks and distributed computing for scalable analytics and insights generation.</li> <li>10. To develop expertise in deep learning models for unstructured data analysis, including images, text, and time-series.</li> <li>11. To evaluate and deploy data science models with a focus on ethical analytics, performance optimization, and real-time applications.</li> </ol>								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<ol style="list-style-type: none"> <li>1. Design and implement preprocessing and visualization pipelines; construct algorithms for data cleaning, transformation, and feature engineering.</li> <li>2. Develop and evaluate advanced supervised and unsupervised learning models; analyze their effectiveness for analytics tasks.</li> </ol>								

<p>3. Apply deep learning techniques to unstructured data; assess the performance of neural networks in analytics applications.</p> <p>4. Build scalable analytics workflows using big data frameworks; optimize distributed systems for real-time data processing.</p> <p>5. Create end-to-end data science and analytics projects, integrating data acquisition, modeling, visualization, and deployment; evaluate trade-offs in analytics pipelines.</p> <p>6. Implement ethical and responsible AI practices; assess the societal and business impact of analytics solutions in real-world applications.</p>		
<b>Course Contents</b>		
<b>Unit 1</b>	<b>Foundations of Data Science and Analytics</b>	<b>6 Hours</b>
<p>Introduction to Data Science and Analytics: Scope, challenges, and applications in business and technology. Data Preprocessing: Handling missing data, normalization, encoding categorical variables. Feature Engineering: Feature selection, dimensionality reduction (PCA, t-SNE, UMAP). Probability and Statistics for Analytics: Distributions, hypothesis testing, Bayesian methods. Python/R for Analytics: Libraries (NumPy, Pandas, Polars, Matplotlib, Seaborn).</p>		
<b>Case Study#1:</b>	Analyze a real-world dataset (e.g., e-commerce sales data) for preprocessing and visualization. Discuss the impact of feature engineering on predictive analytics.	
<b>Case Study #2:</b>	Evaluate the use of UMAP for visualizing high-dimensional data (e.g., customer behavior data) compared to PCA.	
<b>Unit 2</b>	<b>Supervised and Unsupervised Learning for Analytics</b>	<b>8 Hours</b>
<p>Supervised Learning: Linear regression, logistic regression, SVM, decision trees, random forests, ensemble methods (XGBoost, CatBoost, LightGBM). Unsupervised Learning: K-means, hierarchical clustering, DBSCAN, spectral clustering. Model Evaluation: Cross-validation, ROC-AUC, precision-recall, SHAP values for interpretability. Hyperparameter Tuning: Grid search, Bayesian optimization.</p>		
<b>Case Study#3:</b>	Compare ensemble methods (e.g., CatBoost vs. LightGBM) for a predictive analytics task (e.g., customer lifetime value prediction). Discuss interpretability and scalability.	
<b>Case Study#4:</b>	Analyze clustering techniques (e.g., DBSCAN vs. spectral clustering) for market segmentation in a retail analytics scenario.	
<b>Unit 3</b>	<b>Deep Learning for Advanced Analytics</b>	<b>8 Hours</b>
<p>Neural Networks: Perceptrons, activation functions, backpropagation. Deep Learning Frameworks: TensorFlow, PyTorch, JAX. Convolutional Neural Networks (CNNs): Applications in image analytics. Recurrent Neural Networks (RNNs): LSTM, GRU for time-series analytics. Transfer Learning: Pre-trained models (e.g., EfficientNet, BERT). AutoML for analytics.</p>		
<b>Case Study#5:</b>	Evaluate a CNN model for image-based analytics (e.g., defect detection in manufacturing). Discuss the role of transfer learning in reducing training time.	
<b>Case Study#6:</b>	Analyze LSTM models for real-time time-series analytics (e.g., IoT sensor data). Discuss challenges in streaming data analytics.	
<b>Unit 4</b>	<b>Big Data and Real-Time Analytics</b>	<b>6 Hours</b>

Big Data Fundamentals: Characteristics, challenges, and tools. Distributed Computing: Hadoop, Spark, Dask, Ray. Real-Time Data Processing: Apache Kafka, Flink for streaming analytics. Scalable Machine Learning: Distributed training with Spark MLlib, Ray Train. Cloud Platforms: AWS, Google Cloud, Azure for analytics pipelines.

<b>Case Study#7:</b>	Design a real-time analytics pipeline using Apache Kafka and Spark for social media sentiment analysis. Evaluate latency and scalability.
----------------------	---

<b>Case Study#8:</b>	Analyze the trade-offs of edge computing vs. cloud-based analytics for IoT applications (e.g., smart cities).
----------------------	---

<b>Unit 5</b>	Advanced Analytics with Deep Learning	<b>6 Hours</b>
---------------	---------------------------------------	----------------

Generative Models: GANs, VAEs, diffusion models for synthetic data generation. Natural Language Processing (NLP): Transformers, attention mechanisms, large language models. Reinforcement Learning: Deep Q-learning, policy gradients for decision-making analytics. Model Optimization: Quantization, pruning, knowledge distillation.

<b>Case Study#9:</b>	Study diffusion models for generating synthetic customer data in analytics. Discuss ethical considerations in synthetic data usage.
----------------------	---

<b>Case Study#10:</b>	Evaluate transformers for text analytics (e.g., customer feedback analysis). Compare with traditional NLP methods.
-----------------------	--

<b>Unit 6</b>	Ethics and Deployment in Data Science and Analytics	<b>6 Hours</b>
---------------	---	----------------

Ethical AI and Analytics: Bias mitigation, fairness, explainability. Model Deployment: Containerization (Docker, Kubernetes), REST APIs, MLOps. Model Monitoring: Drift detection, real-time performance tracking. Data Privacy: GDPR, differential privacy, federated learning. Responsible AI: Case studies in ethical analytics.

<b>Case Study#11:</b>	Evaluate the deployment of an analytics model in a financial application (e.g., credit scoring). Discuss fairness and regulatory compliance.
-----------------------	--

<b>Case Study#12:</b>	Analyze model drift in a real-time recommendation system (e.g., e-commerce). Discuss strategies for automated retraining and monitoring.
-----------------------	--

**Total Hours: 36**

**Text Books:**

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", 3rd Edition, O'Reilly Media, 2022, ISBN: 978-1098125974
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016, ISBN: 978-0262035613
3. Matei Zaharia, Holden Karau, "Learning Spark: Lightning-Fast Big Data Analysis", 3rd Edition, O'Reilly Media, 2024, ISBN: 978-1492050018

**Reference Books:**

1. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006, ISBN: 978-0387310732
2. Chip Huyen, "Designing Machine Learning Systems", 2nd Edition, O'Reilly Media, 2024, ISBN: 978-1098150556

3. Max Kuhn, Kjell Johnson, "Feature Engineering and Selection: A Practical Approach for Predictive Models", CRC Press, 2019, ISBN: 978-1138079229

**E-Books/Online Learning Resources:**

1. "Deep Learning" by Ian Goodfellow et al., <https://www.deeplearningbook.org/>
2. <https://www.tensorflow.org/tutorials>
3. <https://spark.apache.org/docs/latest/>
4. <https://www.polars.io/docs/>

**Online Courses:**

1. Deep Learning Specialization
  - a. Instructor: Andrew Ng
  - b. Covers deep learning fundamentals and applications.
  - c. Link: <https://www.coursera.org/specializations/deep-learning>
2. Big Data and Real-Time Analytics
  - a. Instructor: University of California, San Diego
  - b. Covers big data tools and streaming analytics.
  - c. Link: <https://www.coursera.org/specializations/big-data>
3. MLOps for Data Science and Analytics
  - a. Instructor: Andrew Ng, Robert Crowe
  - b. Covers model deployment and MLOps for analytics.
  - c. Link: <https://www.coursera.org/specializations/machine-learning-engineering-for-production-mlops>

**The CO-PO Mapping Matrix**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	3	2	-	-	-	-	-	2
CO2	3	3	3	2	2	-	-	-	-	-	2
CO3	2	3	3	2	3	-	-	-	-	-	2
CO4	2	2	3	3	3	-	-	-	-	-	2
CO5	3	3	3	2	3	-	-	-	-	1	3
CO6	2	2	2	2	2	3	-	3	-	-	3



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Cryptography and Network Security			<b>Code</b>	DP04020125107			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	--	90	--	60	--	150

**Prerequisite:**

1. Introduction to computer networks.
2. Computer Network Security.

**Other Courses in this Series:**

1. Advanced Computer Networks & Security

**Course Objectives:**

1. Understand fundamental security principles and cryptographic goals such as confidentiality, integrity, and authentication.
2. Apply core mathematical concepts (e.g., modular arithmetic, number theory) to analyze cryptographic algorithms.
3. Evaluate symmetric and asymmetric cryptographic techniques, including block/stream ciphers and public key systems.
4. Implement and assess authentication mechanisms, including hash functions, MACs, and digital signature schemes.
5. Analyze network and system security architectures, including firewalls, intrusion detection systems, VPNs, and secure protocols.
6. Explore real-world applications and emerging trends in cryptography, such as email/web security, privacy-preserving techniques, and quantum-safe cryptography

**Course Outcomes:****On completion of the course, students will be able to:**

- CO1:** Explain the principles of cryptography and security services, including threats, attacks, and OSI security architecture.
- CO2:** Demonstrate the use of mathematical techniques like modular arithmetic and number theory in designing cryptographic algorithms.

<p><b>CO3:</b> Implement and compare symmetric and asymmetric encryption methods, including AES, DES, RSA, and ECC.</p> <p><b>CO4:</b> Apply cryptographic hash functions and digital signatures for data integrity and authentication.</p> <p><b>CO5:</b> Evaluate network and system security mechanisms, such as VPNs, firewalls, SSL/TLS, and IDS.</p> <p><b>CO6:</b> Analyze and apply security solutions to real-world scenarios, including secure email, web communication, and emerging privacy technologies.</p>		
<b>Course Contents</b>		
<b>Unit 1</b>	<b>Security Foundations &amp; Cryptographic Mathematics</b>	<b>6 Hours</b>
Security goals: confidentiality, integrity, authentication, non-repudiation, Threat model: attacks, services, mechanisms, OSI security architecture. Number theory essentials: modular arithmetic, Euclid's algorithm, Fermat's/Euler's theorems, finite fields, Chinese remainder theorem.		
<b>Case Study#1:</b>	<p><b>Case Study: Moonlight Maze (1999)</b></p> <p>A longstanding cyber-espionage campaign targeted U.S. government networks (e.g., NASA, DoE) using IP spoofing and backdoor infiltration. Detection of anomalous behaviour led to the deployment of advanced IDS tools and adoption of zero-trust architectures, highlighting how threat models drive security design.</p>	
<b>Unit 2</b>	<b>Classical &amp; Symmetric Cryptography</b>	<b>6 Hours</b>
Symmetric cryptographic systems: substitution and transposition ciphers, steganography, Modern block ciphers and stream ciphers: DES, Triple DES, AES, RC4, BC modes (ECB, CBC, CFB, OFB)		
<b>Case Study#2:</b>	<p><b>Case Study: Breaking DES (1997)</b></p> <p>A public challenge by RSA Security cracked DES encryption using brute force. This attack prompted the transition to AES and adoption of longer key sizes, illustrating why algorithm strength and key length matter</p>	
<b>Unit 3</b>	<b>Public-Key Cryptography &amp; Key Management</b>	<b>6 Hours</b>
Public-key crypto fundamentals: RSA, ElGamal, Diffie-Hellman, Elliptic Curve Cryptography, Key management frameworks: symmetric key distribution, PKI, X.509 certificates, Kerberos		
<b>Case Study#3:</b>	<p><b>Case Study: SSL/TLS Handshake Using RSA &amp; AES</b></p> <p>IBM researchers analysed systems where RSA secures key exchange and AES encrypts session data. Common in HTTPS, this hybrid approach balances strong key distribution (RSA) with efficient symmetric encryption (AES) for privacy and performance</p>	
<b>Unit 4</b>	<b>Cryptographic Hashes, MACs &amp; Digital Signatures</b>	<b>6 Hours</b>
Hash functions and MACs: MD5, SHA-1, SHA-256/512, HMAC, CMAC, Digital signature schemes: RSA-based, DSA/DSS, ElGamal, Schnorr		
<b>Case Study#4:</b>	<b>Case Study: SHA-256 &amp; RSA in Healthcare EHR Systems</b>	

	A hospital implemented SHA-256 for integrity checking, RSA for authentication, and TLS for secure communication—ensuring EHR confidentiality and compliance with regulations like HIPAA	
<b>Unit 5</b>	<b>Network &amp; System Security Mechanisms</b>	<b>6 Hours</b>
Network-level security: IPsec (AH/ESP), VPNs (PPTP, L2TP, SSL VPN), SSL/TLS, SET, System-level defence: firewalls (types, architectures), IDS/IPS, malware, buffer overflows, intrusion detection		
<b>Case Study#5:</b>	<b>Case Study: IPsec VPN at University of Namibia (2020)</b> To reduce dependency on costly third-party MPLS VPN, the university built an in-house IPsec tunnel. The setup offered encryption, authentication, lower costs, and comparable performance, showing how network-layer encryption safeguards institutional networks	
<b>Unit 6</b>	<b>Application-Layer &amp; Emerging Topics</b>	<b>6 Hours</b>
Email and web security: PGP, S/MIME, HTTPS, DNSSEC, Privacy enhancements: anonymization (k-anonymity, l-diversity), differential privacy, Optional: lightweight/quantum crypto overview (ASCON, ECC, neural crypto)		
<b>Case Study#6:</b>	<b>Case Study: Quantum-Safe IPsec VPN for JPMorgan Chase (2024)</b> Using quantum key distribution (QKD) over fiber, JPMorgan Chase built a 100 Gbps quantum-safe IPsec tunnel. It refreshed AES-256 keys via QKD—demonstrating cutting-edge integration of quantum cryptography into mainstream secure communications	
		<b>Total Hours: 36</b>
<b>Text Books:</b>		
1. <b>Cryptography &amp; Network Security</b> by Williams Stallings		
<b>Reference Books:</b>		
1. Introduction to Modern Cryptography", Jonathan Katz and Yehuda Lindell, CRC Press		
2. Network Security: Private Communication in a Public World", Charlie Kaufman, Radia Perlman, Mike Speciner, Pearson		
<b>E-Books/Online Learning Resources:</b>		
1. "The Laws of Cryptography with Java Code" – Neal Koblitz Link: <a href="https://cseweb.ucsd.edu/~mihir/cse207/koblitz.pdf">https://cseweb.ucsd.edu/~mihir/cse207/koblitz.pdf</a>		
2. <b>Introduction to Cryptography</b> – Christof Paar & Jan Pelzl <a href="https://www.crypto-textbook.com/">https://www.crypto-textbook.com/</a>		
<b>Online Courses:</b>		
1. Coursera – "Cryptography I" <b>Instructor:</b> Prof. Dan Boneh (Stanford University) Link: <a href="https://www.coursera.org/learn/crypto">https://www.coursera.org/learn/crypto</a>		
2. NPTEL – Cryptography and Network Security Link: <a href="https://nptel.ac.in/courses/106105031">https://nptel.ac.in/courses/106105031</a>		

The CO-PO Mapping Matrix											
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	1	1	1	-	2	-	1	2
CO2	3	3	2	2	1	-	-	-	-	-	2
CO3	2	3	3	2	3	-	-	-	-	1	2
CO4	2	2	2	3	3	-	-	2	-	1	2
CO5	3	3	2	3	3	1	-	2	-	1	2
CO6	2	2	2	2	2	2	-	2	-	2	3



<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>		I		
<b>Course</b>	Distributed Systems			<b>Code</b>		DP04020125108		
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	--	90	--	60	--	150
<b>Prerequisite:</b>								
1. Basics of Computers								
<b>Other Courses in this Series:</b>								
1. Problem Solving through C Programming Lab								
<b>Course Objectives:</b>								
1. To understand the Fundamental Concepts of Distributed Systems.								
2. To evaluate communication mechanism in distributed environments.								
3. To analyse middleware, operating System Support, and security Protocols.								
4. To apply Naming and Directory Services for Resource Identification and Security.								
5. Compare and Apply Consistency Models, Replication, and Fault Tolerance Techniques.								
6. Design and analyse Distributed File Systems and Algorithms								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<b>CO1: Explain the fundamentals, goals, and types of distributed systems and communication mechanisms like IPC, RPC, and multicast.</b>								
<b>CO2: Analyse distributed object communication, remote invocation, and apply security techniques like cryptography and digital signatures.</b>								
<b>CO3: Implement and evaluate naming schemes and directory services (DNS, LDAP) and apply secure communication protocols in distributed systems.</b>								
<b>CO4: Compare and implement consistency models, replication strategies, and fault-tolerant mechanisms</b>								
<b>CO5: Design distributed file systems and apply distributed algorithms such as Chandy-Lamport, load balancing, and gossip-based communication. Demonstrate understanding of distributed databases, cloud computing platforms.</b>								
<b>Course Contents</b>								
<b>Unit 1</b>	<b>Introduction and Communication in Distributed Systems</b>						<b>6 Hours</b>	
<b>Introduction to Distributed Systems: Definition, Goals, Characteristics, Types of Distributed Systems: Examples</b>								

(GFS, Hadoop, IoT, etc.), Design Issues and Challenges, Communication Mechanisms: Inter-process Communication (IPC), Remote Procedure Calls (RPC), Remote Method Invocation (RMI), Message-Oriented Communication, Group Communication and Multicast.		
<b>Case Study#1:</b>	<b>Google File System (GFS)</b>	
<b>Case Study#2:</b>	<b>Case Study: Remote Procedure Call (RPC) in Apache Thrift</b> Demonstrates the use of Apache Thrift to implement Remote Procedure Calls across heterogeneous systems, showcasing how distributed systems achieve interoperability and seamless communication between services.	
<b>Unit 2</b>	<b>Distributed Objects and Process</b>	<b>6 Hours</b>
Distributed objects and remote invocation, Communication between distributed objects, Remote procedure call, Events and notifications - The operating system layer, Protection, Processes and Threads, Communication and invocation, OS Architecture. Security techniques, Cryptographic algorithms, Access control, Digital signatures, Cryptography pragmatics, Needham-Schroeder, Kerberos, Securing electronics transaction, IEEE 802.11 WiFi.		
<b>Case Study#3:</b>		
<b>Case Study#4:</b>		
<b>Unit 3</b>	<b>Naming, Directory Services, and Security</b>	<b>6 Hours</b>
<b>Naming Concepts:</b> Name Resolution, Name Spaces, <b>Distributed Naming Services:</b> DNS, LDAP, Directory Services and Naming Scalability, <b>Security in Distributed Systems:</b> Authentication, Authorization, Confidentiality, Integrity Cryptographic Protocols, Secure Communication and Access Control Security in Web-Based Distributed Systems		
<b>Case Study#5:</b>	<b>Domain Name System (DNS) as a Distributed Naming Service:</b> Understand how DNS operates as a scalable and distributed naming system that maps human-readable domain names to IP addresses.	
<b>Case Study#6:</b>	<b>Securing Distributed Systems using Public Key Infrastructure (PKI):</b> Examine how PKI provides authentication, confidentiality, and data integrity in distributed environments through secure key management.	
<b>Unit 4</b>	<b>Consistency, Replication, and Fault Tolerance</b>	<b>6 Hours</b>
<b>Data-centric Consistency Models:</b> Strict, Sequential, Causal, FIFO <b>Client-centric Consistency Models:</b> Eventual, Monotonic, <b>Replication and Replica Management, Fault Tolerance, Fault Models:</b> Crash, Byzantine <b>Recovery Techniques:</b> Checkpointing, <b>Logging Distributed Commit Protocols:</b> 2PC, 3PC		
<b>Case Study#7:</b>	<b>Consistency and Fault Tolerance in Amazon DynamoDB:</b> Analyse how Amazon DynamoDB, a distributed NoSQL database, ensures high availability and fault tolerance while offering configurable consistency models.	
<b>Case Study#8:</b>	<b>Fault Tolerance and Data Replication in Hadoop HDFS :</b> Explore how Hadoop Distributed File System (HDFS) ensures fault tolerance and data reliability	

	through replication and failure recovery mechanisms.	
<b>Unit 5</b>	<b>Distributed File Systems and Algorithms</b>	<b>6 Hours</b>
<b>Distributed File System Concepts:</b> File Models, Access Semantics Architecture and Design <b>Distributed Algorithms:</b> Snapshot Algorithms (Chandy-Lamport) Deadlock Detection and Resolution Load Balancing, Resource Allocation Gossip Algorithms		
<b>Case Study#9:</b>	<b>Case Studies:</b> <b>Explore how HDFS implements file models, access semantics, and architectural design to handle large-scale data storage.</b>	
<b>Case Study#10:</b>	Explore how distributed algorithms such as the <b>Chandy-Lamport Snapshot, Deadlock Detection, Gossip Algorithms, and Load Balancing</b> are applied in managing consistency, coordination, and resource distribution.	
<b>Unit 6</b>	<b>Advanced Distributed System</b>	<b>6 Hours</b>
Distributed Databases, <b>Cloud Computing:</b> Overview of Platforms, Amazon EC2, Google Cloud, Blockchain and Distributed Ledger Technology, Microservices and Container Orchestration (e.g., Docker, Kubernetes), Emerging Trends in Distributed Systems		
<b>Case Study#11:</b>	Understand how microservices architecture and container orchestration tools like Docker and Kubernetes are used to deploy and scale distributed applications.	
<b>Case Study#12:</b>	Examine how blockchain technology provides fault-tolerant, tamper-proof, and transparent distributed ledgers in real-world systems.	
<b>Total Hours: 36</b>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Andrew S. Tanenbaum &amp; Maarten Van Steen – <i>Distributed Systems: Principles and Paradigms</i></li> <li>2. George Coulouris et al. – <i>Distributed Systems: Concepts and Design</i></li> <li>3. Pradeep K. Sinha – <i>Distributed Operating Systems</i></li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Coulouris et al. – "Distributed Systems :Concepts &amp; Design"</li> <li>2. William Stallings- "Cryptography and Network Security"</li> <li>3. Kenneth Birman- "Reliable Distributed Systems"</li> </ol>		
<b>E-Books/Online Learning Resources:</b>		
<ol style="list-style-type: none"> <li>1. <a href="https://www.pearson.com/store/p/distributed-systems-concepts-and-design/P100000215920">https://www.pearson.com/store/p/distributed-systems-concepts-and-design/P100000215920</a></li> <li>2. <a href="https://www.cse.iitb.ac.in/~mythili/teaching/cs602/slides/book.pdf">https://www.cse.iitb.ac.in/~mythili/teaching/cs602/slides/book.pdf</a></li> <li>3. <a href="https://ocw.mit.edu/courses/6-824-distributed-computer-systems-engineering-spring-2006/">https://ocw.mit.edu/courses/6-824-distributed-computer-systems-engineering-spring-2006/</a></li> <li>4. <a href="https://books.google.com/books/about/Cryptography_and_Network_Security.html?id=3RtBDwAAQBAJ">https://books.google.com/books/about/Cryptography_and_Network_Security.html?id=3RtBDwAAQBAJ</a></li> </ol>		
<b>Online Courses:</b>		
<ol style="list-style-type: none"> <li>1. Distributed Systems <ol style="list-style-type: none"> <li>a. Instructor: Prof. P. K. Biswas (IIT Kharagpur)</li> </ol> </li> </ol>		

- b. Covers Distributed System concepts
- c. Link: <https://nptel.ac.in/courses/106105036>
- 2. Distributed Operating System
  - a. Instructor: Prof. Kavi Arya (IIT Bombay)
  - b. Link: <https://nptel.ac.in/courses/106101063>
- 3. Computer Networks and Internet Protocol
  - a. Instructor:
  - b. Covers DNS, addressing, routing, and naming
  - c. <https://nptel.ac.in/courses/106105183>
- 4. Blockchain Architecture Design and Use Cases
  - a. Instructor: Prof. Sandeep Shukla (IIT Kanpur)
  - b. Covers Blockchain Architecture Design and Use Cases
  - c. Link: <https://nptel.ac.in/courses/106104220>

The CO-PO Mapping Matrix

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2	1	1	-	-	-	-	1	2
CO2	3	3	2	2	2	-	-	1	-	1	2
CO3	3	3	3	2	2	-	-	1	-	1	2
CO4	2	3	3	2	2	-	-	-	-	-	2
CO5	2	3	3	2	2	-	-	-	1	1	2
CO6	2	2	3	2	3	1	-	1	1	2	3



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Human Computer Interface			<b>Code</b>	DP04020125109			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	--	90	--	60	--	150
<b>Prerequisite:</b>								
<ol style="list-style-type: none"> <li>1) Basic understanding of Programming and Data Structures</li> <li>2) Fundamentals of Operating Systems and Computer Graphics</li> </ol>								
<b>Other Courses in this Series:</b>								
<ol style="list-style-type: none"> <li>1) Advanced Computer Architecture</li> </ol>								
<b>Course Objectives:</b>								
<ol style="list-style-type: none"> <li>1. To introduce the principles and practices of Human-Computer Interaction (HCI)..</li> <li>2. To explore the cognitive, social, and emotional aspects of user interaction with computing systems..</li> <li>3. To design intuitive, accessible, and effective user interfaces.</li> <li>4. To equip students with knowledge of usability evaluation and prototyping tools.</li> <li>5. To understand emerging trends in HCI such as AR/VR, gesture interfaces, and brain-computer interaction.</li> </ol>								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<b>CO1: Understand</b> foundational theories of human-computer interaction								
<b>CO2: Apply</b> HCI design principles for developing user-centric systems.								
<b>CO3: Use</b> evaluation techniques to assess usability of interactive systems.								
<b>CO4: Create</b> low-fidelity and high-fidelity prototypes.								
<b>CO5: Analyze</b> and integrate modern trends in HCI like wearable and touchless interfaces.								
<b>Course Contents</b>								
<b>Unit 1</b>	<b>Introduction to HCI</b>							<b>6 Hours</b>
Definition and scope of HCI, HCI and usability goals, User-centered design process, Interdisciplinary nature of								

HCI, Usability, Types of Usability, User Interface (UI), Measurable Human Factors, Accessibility, Differently-abled Users.		
<b>Case Study#1:</b>	Evolution of User Interfaces in Operating Systems- Examine the transition from command-line to graphical and then to voice and touch-based interfaces in systems like Windows, macOS, and Android.	
<b>Case Study#2:</b>	<b>Design Thinking in HCI: Case of Apple vs. Android</b> Analyze how human-centered design philosophy shaped the UX strategies of iOS and Android platforms.	
<b>Unit 2</b>	<b>Human Factors and Cognition</b>	<b>6 Hours</b>
Human perception and memory, Models of interaction: Norman's model, Abowd and Beale framework, Cognitive models: GOMS, KLM, Mental Models, Ergonomics in interface design.		
<b>Case Study#3:</b>	Design of Cockpit Interfaces in Aviation- Analyze how cognitive load, attention, and perception are considered in designing interfaces for pilots, with focus on error reduction and safety.	
<b>Case Study#4:</b>	Medical Device UI – Cognitive Load and Alert Fatigue - Explore how complex UIs in ICU monitors or infusion pumps affect cognitive performance of healthcare worker	
<b>Unit 3</b>	<b>Interaction Styles and Design Principles</b>	<b>6 Hours</b>
Command line, menu, form, direct manipulation, and natural language interfaces, Design principles: Consistency, feedback, affordance, error prevention, Guidelines and heuristics (Nielsen's Heuristics)		
<b>Case Study#5:</b>	ATM Interface Usability Failures- Evaluate usability issues in ATMs and propose redesigns based on Nielsen's Heuristics and interaction design principles.	
<b>Case Study#6:</b>	Self-Service Kiosks: Fast Food Chains (e.g., McDonald's, KFC)-Analyze user flow and feedback mechanisms to recommend design improvements.	
<b>Unit 4</b>	<b>Interface Design and Prototyping</b>	<b>6 Hours</b>
Interface components and layout, Wireframing, storyboarding, and prototyping, Tools for interface design (Figma, Balsamiq, Adobe XD), Mobile-first and responsive design		
<b>Case Study#7:</b>	Redesigning a Government e-Service Portal- Use tools like Figma or Adobe XD to prototype a more usable, responsive version of a public service website (e.g., Indian Railway, DigiLocker).	
<b>Case Study#8:</b>	Mobile App Prototyping for Elderly Users-Design a healthcare app with considerations for age-related cognitive and motor challenges.	
<b>Unit 5</b>	<b>Usability Engineering and Evaluation</b>	<b>6 Hours</b>
Usability testing methods (lab and field), Heuristic evaluation and cognitive walkthroughs, Metrics for usability (efficiency, effectiveness, satisfaction), A/B testing, eye tracking, and analytics		
<b>Case Study#9:</b>	Usability Testing of a Health Monitoring App- Plan and conduct heuristic evaluations, user testing, and analysis for an existing fitness or telemedicine mobile application.	

<b>Case Study#10:</b>	Eye-Tracking Analysis of E-Commerce Websites - Study user attention patterns and layout effectiveness on platforms like Amazon or Flipkart.	
<b>Unit 6</b>	<b>Advanced HCI</b>	<b>6 Hours</b>
Touchless interaction, gesture-based UI, and wearables, Brain-Computer Interfaces (BCIs), Conversational UI and chatbots, Ethics, accessibility, and inclusive design		
<b>Case Study#11:</b>	Gesture-based Interfaces in Smart Classrooms using Kinect-Analyze real-world deployment and challenges in using gesture controls in educational technology.	
	Ethics and Privacy in Brain-Computer Interfaces (BCI)-Explore how BCI systems like Neuralink pose ethical dilemmas related to consent, data security, and control.	
<b>Total Hours: 36</b>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, Human-Computer Interaction, Pearson Education.</li> <li>2. Ben Shneiderman, Catherine Plaisant, Designing the User Interface, Pearson.</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Donald Norman, The Design of Everyday Things, Basic Books.</li> <li>2. Jeff Johnson, Designing with the Mind in Mind, Morgan Kaufmann.</li> <li>3. Jenny Preece, Interaction Design: Beyond Human-Computer Interaction, Wiley.</li> </ol>		
<b>E-Books/Online Learning Resources:</b>		
<ol style="list-style-type: none"> <li>1. <a href="#">NPTEL :: Computer Science and Engineering - NOC:Design &amp; Implementation of Human-Computer Interfaces</a></li> <li>2. <a href="https://www.interaction-design.org">https://www.interaction-design.org</a></li> <li>3. <a href="https://www.w3.org/WAI/">https://www.w3.org/WAI/</a> (Web Accessibility Initiative)</li> <li>4. <a href="#">NPTEL Syllabus - Human-Computer Interaction</a></li> </ol>		
<b>Online Courses:</b>		
<ol style="list-style-type: none"> <li>1. NPTEL: Human-Computer Interaction (IIT Guwahati / IIT Bombay)</li> <li>2. Coursera: Human-Centered Design by University of California</li> <li>3. edX: Human-Computer Interaction by Georgia Tech</li> </ol>		

The CO-PO Mapping Matrix											
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	1	1	1	2	-	-	1	1	-
CO2	3	3	2	2	2	2	-	-	1	1	1
CO3	3	2	2	2	2	2	-	-	1	1	1
CO4	2	3	3	2	3	-	-	-	1	2	2
CO5	2	2	2	1	3	2	1	3	1	2	3



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech (Computer Engineering)			<b>Semester</b>	1			
<b>Course</b>	Digital Image Processing			<b>Code</b>	DP04020125111			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>	<b>TEA</b>		<b>Total</b>	
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
3	3	--	-	90	--	60	--	150
<b>Prerequisite:</b>								
<ol style="list-style-type: none"> <li>1. Fundamentals of Linear Algebra and Calculus</li> <li>2. Basics of Programming (Python/C++ preferred)</li> <li>3. Signals and Systems</li> <li>4. Probability and Statistics</li> </ol>								
<b>Other Courses in this Series:</b>								
<b>Course Objectives:</b>								
<b>CO1:</b> To understand the fundamentals of digital image processing, including image formation, representation, and enhancement techniques.								
<b>CO2:</b> To enable students to apply spatial and frequency domain techniques for image processing and analysis.								
<b>CO3:</b> To introduce advanced image processing algorithms for segmentation, feature extraction, and object recognition.								
<b>CO4:</b> To explore deep learning techniques for image processing tasks such as classification and detection.								
<b>CO5:</b> To develop skills in implementing image processing pipelines for real-world applications like medical imaging and computer vision.								
<b>CO6:</b> To evaluate the performance and ethical implications of image processing systems in practical scenarios.								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<ol style="list-style-type: none"> <li>1. <b>CO1:</b> Analyze digital images and apply preprocessing techniques; construct algorithms for image enhancement and restoration.</li> <li>2. <b>CO2:</b> Implement spatial and frequency domain techniques; evaluate their effectiveness for image transformation and filtering.</li> <li>3. <b>CO3:</b> Design and apply image segmentation and feature extraction algorithms; analyze their performance in</li> </ol>								

<p>real-world applications.</p> <p>4. <b>CO4:</b> Develop deep learning models for image classification, detection, and segmentation; assess their accuracy and computational efficiency.</p> <p>5. <b>CO5:</b> Create end-to-end image processing pipelines; evaluate trade-offs in performance, scalability, and robustness.</p> <p>6. <b>CO6:</b> Implement ethical image processing practices; assess the societal impact of image processing applications in domains like surveillance and healthcare.</p>		
<b>Course Contents</b>		
<b>Unit 1</b>	<b>Fundamentals of Digital Image Processing</b>	<b>6 hours</b>
<p>Introduction to Digital Image Processing: Image formation, digitization, and representation. Image Types: Grayscale, color, binary images. Image Sampling and Quantization. Basic Image Operations: Pixel-based operations, histogram processing. Image Enhancement: Contrast stretching, histogram equalization. Python/C++ for Image Processing: Libraries (OpenCV, PIL, scikit-image).</p>		
<b>Case Study#1:</b>	<p><i>Case Study #1:</i> Analyze the effect of histogram equalization on low-contrast medical images (e.g., X-ray images). Discuss improvements in diagnostic visibility.</p> <p><i>Case Study #2:</i> Evaluate the impact of sampling and quantization on image quality in a real-world application (e.g., satellite imagery).</p>	
<b>Unit 2</b>	<b>Spatial and Frequency Domain Techniques</b>	<b>6 Hours</b>
<p>Spatial Domain Processing: Point processing, neighborhood operations, spatial filtering (smoothing, sharpening). Convolution and Correlation. Frequency Domain Processing: Fourier Transform, Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT). Frequency Domain Filters: Low-pass, high-pass, band-pass filters. Image Restoration: Noise models, inverse filtering, Wiener filtering.</p>		
<b>Case Study#2:</b>	<p><i>Case Study #3:</i> Compare the performance of spatial vs. frequency domain filters for denoising a corrupted image (e.g., Gaussian noise in MRI scans).</p> <p><i>Case Study #4:</i> Analyze the use of FFT for edge detection in industrial quality control images.</p>	
<b>Unit 3</b>	<b>Image Segmentation and Feature Extraction</b>	<b>6 Hours</b>
<p>Image Segmentation: Thresholding, region-based segmentation, edge-based segmentation, watershed algorithm. Feature Extraction: Edge detection (Canny, Sobel), corner detection (Harris), SIFT, SURF. Morphological Operations: Dilation, erosion, opening, closing. Shape Analysis: Contours, moments, and shape descriptors.</p>		
<b>Case Study#3:</b>	<p><i>Case Study #5:</i> Evaluate the watershed algorithm for segmenting cells in microscopic images. Discuss challenges in handling overlapping objects.</p> <p><i>Case Study #6:</i> Analyze the use of SIFT features for object recognition in autonomous vehicle systems.</p>	
<b>Unit 4</b>	<b>Deep Learning for Image Processing</b>	<b>6 Hours</b>
<p>Introduction to Deep Learning: Neural networks, CNN architecture, activation functions. Deep Learning</p>		

Frameworks: TensorFlow, PyTorch. Convolutional Neural Networks (CNNs): Layers, pooling, architectures (VGG, ResNet). Transfer Learning: Pre-trained models (e.g., ResNet, EfficientNet). Image Classification and Object Detection: YOLO, Faster R-CNN.		
<b>Case Study#4:</b>	<p><i>Case Study #7:</i> Evaluate a CNN model for classifying medical images (e.g., cancer detection in mammograms). Discuss the role of transfer learning in improving accuracy.</p> <p><i>Case Study #8:</i> Analyze the performance of YOLO for real-time object detection in surveillance systems.</p>	
<b>Unit 5</b>	<b>Advanced Image Processing Techniques</b>	<b>6 Hours</b>
Image Restoration and Reconstruction: Super-resolution, inpainting, deblurring. 3D Image Processing: Volumetric data, medical imaging (CT, MRI). Multi-spectral and Hyperspectral Imaging. Generative Models for Images: GANs, diffusion models for image synthesis. Video Processing: Motion estimation, optical flow.		
<b>Case Study#5:</b>	<p><i>Case Study #9:</i> Study the application of GANs for generating synthetic medical images. Discuss ethical implications of synthetic image usage.</p> <p><i>Case Study #10:</i> Evaluate super-resolution techniques for enhancing low-resolution CCTV footage.</p>	
<b>Unit 6</b>	<b>Ethics and Applications of Image Processing</b>	<b>6 Hours</b>
Ethical Considerations: Bias in image processing, privacy in surveillance, fairness in facial recognition. Real-World Applications: Medical imaging, autonomous vehicles, augmented reality. Image Processing Pipelines: End-to-end workflows, deployment with APIs. Performance Evaluation: Metrics (PSNR, SSIM, IoU). Responsible AI: Mitigating biases in image-based systems.		
<b>Case Study #6</b>	<p><i>Case Study #11:</i> Evaluate the deployment of a facial recognition system in a public safety application. Discuss privacy and bias concerns.</p> <p><i>Case Study #12:</i> Analyze the use of image processing in autonomous driving (e.g., lane detection). Discuss robustness and safety considerations.</p>	
<b>Total Hours:36</b>		
<b>Text Books:</b>		
<ol style="list-style-type: none"> <li>1. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", 4th Edition, Pearson, 2018, ISBN: 978-0133356724</li> <li>2. Richard Szeliski, "Computer Vision: Algorithms and Applications", 2nd Edition, Springer, 2022, ISBN: 978-3030343729</li> <li>3. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", 3rd Edition, O'Reilly Media, 2022, ISBN: 978-1098125974</li> </ol>		
<b>Reference Books:</b>		
<ol style="list-style-type: none"> <li>1. Anil K. Jain, "Fundamentals of Digital Image Processing", Pearson, 2015, ISBN: 978-9332570320</li> <li>2. Milan Sonka, Vaclav Hlavac, Roger Boyle, "Image Processing, Analysis, and Machine Vision", 4th Edition,</li> </ol>		

Cengage Learning, 2014, ISBN: 978-1133593607

- John C. Russ, "The Image Processing Handbook", 7th Edition, CRC Press, 2015, ISBN: 978-1498740265

**E-Books/Online Learning Resources:**

- "Computer Vision: Algorithms and Applications" by Richard Szeliski, <http://szeliski.org/Book/>
- <https://opencv-python-tutroals.readthedocs.io/en/latest/>
- <https://www.tensorflow.org/tutorials/images>

**Online Courses:**

- Digital Image Processing**
  - Instructor: Prof. P.K. Biswas, IIT Kharagpur
  - Covers fundamentals of image processing and applications.
  - Link: <https://nptel.ac.in/courses/117105135>
- Computer Vision and Image Processing**
  - Instructor: Prof. Jayanta Mukhopadhyay, IIT Kharagpur
  - Covers advanced image processing techniques.
  - Link: <https://nptel.ac.in/courses/106105224>
- Deep Learning for Computer Vision**
  - Instructor: Andrew Ng
  - Covers deep learning techniques for image processing.
  - Link: <https://www.coursera.org/learn/convolutional-neural-networks>

**The CO-PO Mapping Matrix**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	2	1	2	-	-	-	-	-	1
CO2	3	3	3	2	3	-	-	-	-	-	1
CO3	2	3	3	2	3	-	-	-	-	-	2
CO4	2	2	3	2	3	-	-	-	-	-	2
CO5	3	3	3	2	3	-	-	-	-	-	3
CO6	2	2	1	1	2	3	2	3	1	-	2



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M. Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Software and tools for scientific research I			<b>Code</b>	DP04020125113			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
2	--	--	4	--	--	--	100	100
<b>Prerequisite:</b>								
1. Research Methodology & IPR								
<b>Other Courses in this Series:</b>								
2. Software and tools for scientific research II								
<b>Course Objectives:</b>								
<ol style="list-style-type: none"> <li>1. To introduce students to key research databases such as Google Scholar, Scopus, and Web of Science for effective literature review.</li> <li>2. To train students in the use of reference management tools like Zotero and Mendeley for organizing citations and bibliographies.</li> <li>3. To familiarize students with academic profiling platforms such as ORCID, ResearchGate, and Google Scholar for building research identity.</li> <li>4. To develop skills in scientific writing using tools like LaTeX and Microsoft Word, focusing on structure, clarity, and formatting.</li> <li>5. To ensure understanding of plagiarism, academic ethics, and the use of detection tools such as Turnitin and Grammarly.</li> </ol>								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<p><b>CO1:</b> Conduct effective literature reviews using academic databases such as Google Scholar, Scopus, and Web of Science.</p> <p><b>CO2:</b> Organize and manage references using citation tools like Mendeley or Zotero, and generate accurate bibliographies.</p> <p><b>CO3:</b> Create and maintain professional academic profiles on platforms like ORCID, ResearchGate, and Google Scholar.</p> <p><b>CO4:</b> Write and format scientific documents using LaTeX and MS Word, incorporating citations, equations,</p>								

and figures.

**CO5:** Evaluate academic content for originality and ethics using plagiarism detection tools such as Turnitin and Grammarly.

### Suggested List of Laboratory Experiments/Assignments

<b>1</b>	<b>Research Discovery &amp; Journal Quality</b> Use <b>Google Scholar, Scopus, and Web of Science</b> to search, filter, and download high-quality articles.
<b>2</b>	<b>Reference Management with Zotero/Mendeley</b> Install, organize, cite, and generate bibliographies using <b>Zotero</b> or <b>Mendeley</b> .
<b>3</b>	<b>ORCID, ResearcherID &amp; ResearchGate Setup</b> Create and link academic researcher profiles.
<b>4</b>	<b>Plagiarism Detection</b> Analyze a sample research draft using <b>Grammarly</b> and <b>PlagiarismCheck.org</b> or <b>Turnitin (if accessible)</b> .
<b>5</b>	<b>LaTeX for Scientific Writing – Basics</b> Create a basic document with title, sections, and mathematical equations using <b>Overleaf</b> .
<b>6</b>	<b>LaTeX Advanced – Tables, Citations &amp; Figures</b> Integrate images, tables, and reference libraries (.bib) into LaTeX.
<b>7</b>	<b>Research Writing Tools</b> Use <b>Grammarly, QuillBot, and Hemingway Editor</b> for enhancing academic writing.
<b>Survey Paper</b>	<b>Students are required to write a research article suitable for submission to a Scopus-indexed or IEEE conference/journal. The article should demonstrate the effective use of modern research tools such as Zotero/Mendeley (reference management), LaTeX (scientific writing), Grammarly/Turnitin (writing ethics), R/Python/Excel (data analysis), and Scopus/Journal Finder (journal selection).</b>

#### Text Books:

1. How to Write and Publish a Scientific Paper Robert A. Day & Barbara Gastel
2. Scientific Writing and Communication: Papers, Proposals, and Presentations Angelika H. Hofmann
3. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython Wes McKinney

**Reference Books:****1. The Craft of Research**

Wayne C. Booth, Gregory G. Colomb, Joseph M. Williams

**2. LaTeX: A Document Preparation**

System Leslie Lamport

**E-Books/Online Learning Resources:**<https://www.zotero.org/><https://www.mendeley.com/><https://www.overleaf.com/><https://quillbot.com/>**Online Courses:**<https://en.wikibooks.org/wiki/LaTeX>**Software Tools, Simulation Platforms & Hardware Kits:****The CO-PO Mapping Matrix**

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	0	2	0	0	0	0	0	0	3
CO2	0	0	0	0	3	0	0	0	3	2	2
CO3	0	0	0	0	0	0	0	2	3	0	2
CO4	0	0	2	0	3	0	0	0	3	0	0
CO5	0	0	0	0	0	2	3	0	2	0	2



**Dnyaan Prasad Global University**  
**Dr. D. Y. Patil Unitech Society**  
**School of Technology and Research**

<b>Program</b>	M.Tech. (Computer Science & Engineering)			<b>Semester</b>	I			
<b>Course</b>	Environmental Science			<b>Code</b>	DP04020125114			
	<b>Teaching Scheme (Hrs/Week)</b>			<b>Examination Scheme &amp; Marks</b>				
<b>Credits</b>	<b>Lectures</b>	<b>Tutorials</b>	<b>Practical</b>	<b>CA</b>		<b>TEA</b>		<b>Total</b>
				<b>Theory</b>	<b>Practical</b>	<b>Theory</b>	<b>Practical</b>	
2	1	1	--	50	50			100
<b>Prerequisite:</b> Biology, Chemistry, Geography								
<b>Other Courses in this Series: Nil</b>								
<b>Course Objectives:</b>								
<ol style="list-style-type: none"> <li>1. To introduce the basic concepts of environment, ecology and explain the interdependence between living organisms and their surroundings.</li> <li>2. To create awareness about the environmental and health hazards associated with various types of wastes.</li> <li>3. To provide comprehensive knowledge of India's key environmental laws and regulations and ethics.</li> <li>4. To evaluate the environmental impact of digital technologies.</li> <li>5. To understand the role of Green IT policies and international/national standards.</li> </ol>								
<b>Course Outcomes:</b>								
<b>On completion of the course, students will be able to:</b>								
<b>CO1:</b> Understand the basics of Environment, Ecology and Interaction among Living Things.								
<b>CO2:</b> Identify various types of pollution (air, water, soil, noise, thermal, nuclear) and analyze their ecological and health impacts.								
<b>CO3:</b> Apply the principles of environmental ethics and ethical reasoning to technology-related environmental challenges.								
<b>CO4:</b> Integrate AI/ML, Big Data, and GIS technologies to build innovative solutions for real-world environmental challenges.								
<b>CO5:</b> Design and evaluate decision support systems for real-time environmental data analysis and disaster risk reduction.								
<b>Course Contents</b>								

<b>Unit 1</b>	<b>Fundamentals of Environmental Science</b>	<b>6 Hours</b>
Introduction to Environment and Ecology, Ecosystems and Biodiversity, Natural Resources: Renewable & Non-renewable, Climate Change: Causes, Impact & Mitigation, Natural resources: forest, water, mineral, energy, and land. Interdependence between resources and computing.		
<b>Case Study#1:</b>	<b>Biodiversity Monitoring Using AI and Drones:</b> Study and Examine how AI and drone-based monitoring is used in forests (e.g., in the Western Ghats or Sundarbans) to track endangered species and prevent illegal logging.	
<b>Case Study#2:</b>	<b>Ecosystem &amp; Biodiversity Observation-</b> Study and Identify different types of flora and fauna, Note presence of native vs. invasive species, Record biodiversity through photos/sketches.	
<b>Unit 2</b>	<b>Pollution and Waste Management</b>	<b>6 Hours</b>
Types of Pollution Air, Water, Soil, Noise pollution, Thermal nuclear, sources, health/ecological effects., E-waste, hazardous waste: management practices, GIS-enabled pollution monitoring networks.; effects and control measures. Solid and hazardous waste management; e-waste and its handling in IT industries.		
<b>Case Study#3:</b>	<b>E-Waste Management:</b> Evaluate and study how electronic waste is generated across the IT life cycle, Suggest strategies for responsible recycling, reuse, or redesign of electronic products.	
<b>Case Study#4:</b>	<b>Energy Use Estimation:</b> Estimate total daily/weekly energy usage in kWh., Research the equivalent <b>CO<sub>2</sub> emissions generated</b> from that usage.	
<b>Unit 3</b>	<b>Environmental Responsibility: Ethics, Laws, and Governance</b>	<b>6 Hours</b>
Introduction to environmental ethics, Ethical theories and environmental values, National Environmental Policy (India) , Environmental Protection Act, 1986, E-waste Management Rules, 2016, Forest Conservation Act, Air and Water Acts, ISO 14001 (Environmental Management Systems), National and global environmental movements and policy initiatives.		
<b>Case Study#5:</b>	<b>Implementation of ISO 14001 in IT Parks:</b> Explore how major IT campuses (e.g., Infosys, TCS) have implemented ISO 14001: Environmental Management Systems to manage energy, water, and emissions.	
<b>Case Study#6:</b>	<b>Environmental Ethics in Tech Product Design:</b> Study Apple phone's approaches to designing environmentally friendly electronics using recyclable materials and modular components.	
<b>Unit 4</b>	<b>Technology-Driven Approaches to Environmental Solutions</b>	<b>6 Hours</b>
Role of AI/ML in Environmental Monitoring and prediction (air quality, weather, disaster), Big Data for Climate Change Analysis, Geographic Information Systems (GIS) for Environmental Applications, Computational Models for Pollution Control & Disaster Management, Carbon footprint of computing systems.		
<b>Case Study#7:</b>	<b>Flood Simulation Models in Kerala – Post-2018 Disaster"-</b> What computational models (e.g., hydro logical models) were used to simulate flood risks?	

	How can machine learning enhance disaster prediction models? Suggest ways computational models can be integrated into early warning systems.
<b>Case Study#8:</b>	<b>Smart Campus Initiatives using IOT at IIT Madras:</b> What smart technologies were implemented (e.g. automated lighting, temperature control)? How did IoT solutions help conserve energy and reduce carbon footprint? Propose a smart solution for your college campus using IoT.
<b>Unit 5</b>	<b>Smart Systems for Environmental Management and Sustainable Urban Systems</b> <b>6Hours</b>
Smart Cities & IoT-based Environmental Sensing, Remote Sensing & Satellite-Based Environmental Monitoring, Edge Computing for Real-Time Environmental Data Processing. Decision support systems for disaster risk reduction, Green Computing, Green IT policy and standards. National programs: Smart Cities Mission, Digital India Green Initiatives.	
<b>Case Study#9:</b>	<b>How can smart water metering systems contribute to water conservation in urban areas?</b>
<b>Case Study#10:</b>	<b>Pune Smart City – IoT-Based Air Quality Monitoring:</b> Study and Examine how Pune city deployed smart poles and sensors under its Smart Cities Mission to monitor real-time air quality and traffic.
<b>Total Hours: 30</b>	
<b>Field Work/Visit:</b>	
<b>Visit an IT company campus (e.g., Infosys, TCS, Wipro)</b>	
<ul style="list-style-type: none"> <li>• Study green building design, solar panels, and waste management.</li> <li>• Interview the CSR or sustainability team.</li> </ul>	
<b>Conduct interviews with local farmers or cooperatives about supply chain traceability needs.</b>	
<ul style="list-style-type: none"> <li>• Map out the supply chain of a common agricultural product (e.g., rice, wheat, sugarcane).</li> <li>• Survey e-waste collection points and propose a tracking mechanism and solution for it.</li> </ul>	
<b>Visit a water body during or after rainfall:</b>	
<ul style="list-style-type: none"> <li>• Observe and record water flow, drainage blockage, and possible flood indicators.</li> </ul>	
<b>Textbooks:</b>	
1) Benny Joseph, <i>Environmental Studies</i> , McGraw Hill Education, ISBN: 9780070648265 2) Anubha Kaushik & C.P. Kaushik, <i>Perspectives in Environmental Studies</i> , New Age International, ISBN: 9788122433237 3)R.R. Barthwal, <i>Environmental Pollution: Monitoring and Control</i> , New Age International, ISBN: 9788122411778 4)William P. Cunningham and Mary Ann Cunningham, by William P. Cunningham and Mary Ann Cunningham” Principles of Environmental Science: Inquiry and Applications”.	

<b>Reference Books:</b>
<p>1) Erach Bharucha, <i>Textbook of Environmental Studies</i>, 4th Edition, University Press</p> <p>2) Gilbert M. Masters, <i>Introduction to Environmental Engineering and Science</i>, 3rd Edition, Pearson</p> <p>3) Peirce, Weiner, Vesilind, <i>Environmental Pollution and Control</i>, 4th Edition, Elsevier</p> <p>4) Brunner R.C., 1989, <i>Hazardous Waste Incineration</i>, McGraw Hill Inc., 1st Edition, 480p, 1989</p>
<b>E-Books/Online Learning Resources:</b>
<p>1. Website: <a href="https://education.nationalgeographic.org">https://education.nationalgeographic.org</a></p> <p>2. Website: <a href="https://www.unep.org">https://www.unep.org</a></p> <p>3. Website: <a href="https://enviroliteracy.org">https://enviroliteracy.org</a></p> <p>4. Website: <a href="#">Coursera: Environmental Management &amp; Ethics – University of Leeds</a></p>
<b>Online Courses:</b>
<p><b>1) Environmental Science: Concepts, Ecosystems, and Natural Resources</b></p> <p>a. Instructor: Dr. Y.S.C. Khuman &amp; Dr. Jagdamba Prasad (IGNOU)</p> <p>b. Covers Concepts of ecology and ecosystem types and Land, soil &amp; water resource management.</p> <p>c. Link: <a href="#">Swayam - Environmental Studies</a></p> <p><b>2) Sustainability and Environmental Management</b></p> <p>a. Instructor: Harry James</p> <p>b. Covers Environmental science foundations for management, Carbon, climate change concepts,</p> <p>c. Link: <a href="#">Udemy - Environmental Science and Sustainability</a></p> <p><b>3) Environmental Studies: Pollution, EIA, and Biodiversity</b></p> <p>a. Instructor: Prof. Harshit Sosan Lakra (IIT Roorkee)</p> <p>b. Covers Introduction &amp; overview of EIA, Legal frameworks &amp; EIA policies</p> <p>c. Link: <a href="#">NPTEL Environmental Studies</a></p> <p><b>4) Climate Change Science and Predictive Modelling</b></p> <p>a. Instructor : Prof. David Archer (University of Chicago)</p> <p>b. Covers Heat, Light, and Energy, First Climate Model, Greenhouse Gases &amp; the Atmosphere</p> <p>c. Link: <a href="#">Coursera: Global Warming I – The Science and Modelling of Climate Change</a></p> <p><b>5) Global Environmental Challenges and Sustainability Strategies.</b></p> <p>a. Instructor: David Rippin, Ed Garrett, Eleanor Jew, Joshua Kirshner, Ollie Moore (University of York)</p> <p>b. covers Protecting Our Oceans, Environment &amp; Development, Energy &amp; Society.</p> <p>c. Link: <a href="#">Future Learn - Global Environmental Issues</a></p>

The CO-PO Mapping Matrix											
CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	1	-	-	-	2	3	-	-	--	-
CO2	2	2	-	-	-	3	3	-	---	-	--
CO3	1	2	-	--	-	2	3	3	---	-	-
CO4	3	3	1	1	2	2	2	1	-	-	1
CO5	3	3	1	1	2	2	3	1	-	-	2