



HYDERABAD (SIND)
NATIONAL COLLEGIATE BOARD

HSNC Board's

Smt. Chandibai Himathmal Mansukhani College (Autonomous)



(Affiliated to the University of Mumbai)

University College Code : 217-JD Office : T14

Principal : Dr. Manju Lalwani Pathak

Ref. No: CHM (A) AC/02/2026-27

Date: 27th June, 2026

CIRCULAR

The immediate attention of all concerned is invited to this Office Circular No. CHM (A) AC 11/2026 dated 19th June, 2026 regarding Choice Based and Credit Based Syllabus (CBCS), of Smt. CHM College (Autonomous), under the guidelines of University of Mumbai, as per Academic Framework of NEP 2020, for all subjects of T.Y. B.Sc in Computer Science SEM-V and SEM-VI.

This is in continuation with curriculum approved by Academic Council for all the subjects of F.Y. B.Sc in Computer Science (SEM-I), S.Y. B.Sc in Computer Science (SEM-III) and F.Y. B.Sc in Computer Science (SEM-II) & S.Y. B.Sc in Computer Science (SEM – IV) vide Circular Reference Numbers CHM (A) AC/C/01/2025 dated 18th June, 2025, CHM (A) AC/C/01A/2025 dated 21st July 2025 and CHM (A) AC/C/02/2025 dated 20th November, 2025 respectively.

It is hereby communicated that the recommendations of the curriculum designed by the Ad-hoc Board of studies in Computer Science coordinated by the Dean, Faculty of Applied Sciences in the meeting of Academic Council held on 20th June, 2026 have been approved.

In accordance, therewith, the syllabus as per the CBCS, has been brought into force with effect from the Academic Year 2026-27 for T.Y. B.Sc in Computer Science, in continuation with syllabus of F.Y. B.Sc in Computer Science and S.Y. B.Sc in Computer Science of 2025-2026 (updated), and accordingly the same is attached for reference and is available on the College's website www.chmcollege.in

Ulhasnagar – 421003

27th June 2026

Dr. Manju Lalwani Pathak
Principal & Chairperson, Academic Council

Copy forwarded for information to:

1. The Office of Chairperson, Academic Council
2. The Dean, Faculty of Applied Sciences
3. The Chairperson, Ad-hoc Board of Studies
4. The Controller of Examination
5. The Registrar

PREAMBLE

In today's digital age, the influence of computers on how we live and work is profound and far-reaching. Computers have become integral to the fabric of modern civilization. Their widespread application across industries has sparked rapid technological progress, resulting in constantly evolving domains within the computing field. In response, it is essential for students to build a strong foundation that not only supports current competencies but also prepares them to adapt to ongoing advancements.

Aligned with the objectives of the National Education Policy (NEP) 2020, the updated undergraduate Computer Science curriculum aims to equip students with the agility, skills and knowledge required to confidently navigate the dynamic world of technology. Acknowledging that programming languages, tools, and platforms may change over time, the curriculum emphasizes adaptability and resilience. Students will engage with a broad spectrum of programming paradigms, tools, and technologies while gaining a deep understanding of the core principles of Computer Science.

The program's broad spectrum encompasses not only essential key courses such as programming, data structures, computer architecture, algorithms, but also database systems, operating systems, and software engineering. These are complemented by including advanced subjects in areas like artificial intelligence, computer networks, distributed computing, cybersecurity, computer graphics, human-computer interaction, multimedia systems, scientific computing, web technologies and various state-of-the-art disciplines in ever evolving field of computer science.

Key Philosophy the Program:

- **Establish Strong Foundations:** Develop a solid conceptual base in Computer Science.
- **Imbibe Practical Skills:** Cultivate problem-solving, programming, and design capabilities.
- **Embrace Emerging Trends:** Introduce new technologies and concepts in a progressive and coherent manner.
- **Prepare for Industry Demands:** Equip students to confidently face challenges in the ICT sector.

In recognition of the evolving aspirations of our students, this program is designed not only to prepare them for professional careers in the technology sector but also to facilitate opportunities for advanced research. The primary objective is to provide a comprehensive curriculum that integrates both theoretical foundations and practical expertise, equipping graduates with the skills necessary to excel in their professional endeavours while fostering a commitment to lifelong learning.

This program serves as a gateway to a successful career in the software industry while also encouraging students to pursue further academic and research opportunities. Graduates will be well-positioned to transition seamlessly into postgraduate studies in Computer Science, where they may engage in research and development roles, contribute to the IT industry, or explore careers in business management.

As we present this curriculum, we invite students to embark on a journey of intellectual exploration and innovation, ensuring they are not only well-prepared for the present but also strategically positioned to adapt and shape the future of Computer Science.

PROGRAMME OUTCOME (POs)

PO1: Disciplinary Knowledge

Demonstrate a sound understanding of computer science principles, including programming, algorithms, data structures, databases, operating systems, computer networks, software engineering, artificial intelligence, cybersecurity, cloud computing, and emerging technologies to solve computing problems effectively.

PO2: Critical Thinking and Analytical Reasoning

Analyze complex computing problems using computational thinking, logical reasoning, mathematical foundations, and analytical skills to evaluate alternative solutions and make informed technical decisions.

PO3: Problem Solving and Scientific Skills

Apply scientific principles, modern computing techniques, programming skills, and appropriate software tools to design, develop, test, and implement efficient, secure, and scalable computing solutions.

PO4: Research Competence

Apply research methodologies, literature review, experimentation, data analysis, and interpretation to investigate computing problems, evaluate technologies, and develop innovative and evidence-based solutions.

PO5: Digital Literacy

Demonstrate proficiency in modern computing environments, programming platforms, databases, cloud technologies, AI tools, cybersecurity practices, and digital resources while ensuring ethical and responsible use of technology.

PO6: Communication Skills

Communicate effectively through written reports, oral presentations, scientific publications, visual media, and interpersonal interactions with diverse audiences.

PO7: Environmental Sustainability and Community Engagement

Demonstrate empathy, appreciate cultural diversity, engage in community service, and promote environmental sustainability.

PO8: Ethics and Professional Values

Apply professional ethics, legal responsibilities, cybersecurity principles, data privacy practices, and environmental sustainability while developing and deploying computing solutions for the benefit of society.

PO9: Leadership and Teamwork

Work collaboratively in diverse teams, demonstrate leadership qualities, and contribute effectively to achieving common goals.

PO10: Lifelong Education

Engage in self-directed lifelong education for continuous personal and professional development.

PO11: Employability and Entrepreneurship

Demonstrate technical competence, innovation, entrepreneurial thinking, and professional skills required for successful careers, higher education, research, and technology-based entrepreneurship in the global computing industry.



HSNC Board's
Smt. Chandibai Himathmal Mansukhani College, Ulhasnagar
(Autonomous)
Affiliated to the University of Mumbai

Bachelor of Science
(Computer Science)
(Self-Financing Course)

Semester – I

Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2025-2026

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Vertical – 1

Digital Systems and Architecture : 2 Credits

Fundamentals of Database Systems : 2 Credits

Computer Science Practical 1 : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Title: Digital Systems and Architecture

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Digital Systems and Architecture
Course Code: CHMCSI1

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>The Digital Systems and Architecture course offers a fundamental introduction to the key concepts of digital systems and computer architecture. It focuses on the design and organization of digital circuits and systems that underpin today's computing devices.</p> <p>Relevance:</p> <p>In today's fast-paced technological landscape, knowledge of digital systems and architecture is crucial. Digital systems are embedded in everything from smartphones to supercomputers. This course is vital for those aiming to understand how these systems function and play a role in advancing their design and development.</p> <p>Usefulness:</p> <p>The course provides students with the essential knowledge and skills to design, analyze, and optimize digital systems. It also acts as a steppingstone into the broader field of computer architecture, forming a strong foundation for advanced studies and practical applications in modern computing.</p> <p>Application:</p> <p>The knowledge gained in this course has wide-ranging practical applications across areas such as embedded systems, computer networks, signal processing, and more. Students will develop the ability to translate theoretical concepts into practical solutions, effectively bridging the gap between abstract principles and real-world implementation.</p> <p>Interest:</p> <p>Digital Systems and Architecture is an intellectually engaging course that combines theoretical foundations with practical application, capturing students' interest through its dynamic and relevant content. The challenge and excitement of designing efficient, high-performance digital systems often inspire curiosity and enthusiasm among learners.</p> <p>Connection with Other Courses:</p> <p>This course builds essential connections with other areas of computer science, serving as a strong foundation for advanced subjects like computer organization, microprocessor systems,</p>

		<p>and hardware description languages. The concepts learned here contribute to a continuous and cohesive understanding of computer systems.</p> <p>Demand in the Industry:</p> <p>As the demand for faster and more efficient computing systems grows, professionals skilled in digital systems and architecture are increasingly in demand. Industries such as electronics, telecommunications, automotive, and healthcare are actively seeking experts in designing and optimizing digital systems.</p> <p>Job Prospects:</p> <p>Graduates with expertise in digital systems and architecture are well-equipped for a wide range of career opportunities. Potential roles include digital design engineer, embedded systems developer, hardware architect, and systems analyst. The skills gained in this course pave the way for careers in various industries where digital technology is integral to innovation and progress.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: To introduce students to number systems and enable them to perform interconversion among binary, decimal, octal, and hexadecimal systems used in digital systems.</p> <p>CO(A) 2: To develop an understanding of Boolean algebra, logic gates, and various methods for simplifying logic expressions for digital circuit design.</p> <p>CO(A) 3: To impart knowledge of combinational and sequential circuits, their construction, and working principles, including adders, multiplexers, flip-flops, and counters.</p> <p>CO(A) 4: To provide a strong foundation in computer organization and architecture, including memory hierarchy, I/O mechanisms, and processor functions.</p>	

	<p>CO(A) 5: To introduce instruction set architecture, microprocessor operations, pipelining concepts, and compare RISC and CISC paradigms for processor design understanding.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students will be able to –</p> <p>CO1: Understand and apply various number systems and perform accurate conversions between them.</p> <p>CO2: Analyze and simplify digital logic circuits using Boolean algebra and Karnaugh maps</p> <p>CO3: Design and construct combinational and sequential digital circuits including adders, flip-flops, and counters.</p> <p>CO4: Demonstrate understanding of computer architecture concepts, including memory organization, interconnections structures, and I/O methods.</p> <p>CO5: Explain and evaluate instruction sets, addressing modes, and processor functions of 8085/8086 architecture, including concepts of pipelining, ILP, and RISC/CISC.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Basics: Number system, Number system conversion (Binary to other number system, Decimal to other number system, Octal to other number system, Hexadecimal to other number system). • Fundamentals of Digital Logic: Boolean algebra, Logic Gates, Simplification of Logic Circuits: Algebraic Simplification, Karnaugh Maps. • Combinational Circuits: Adders, Subtractors, Multiplexer, De-Multiplexer. • Sequential Circuits: Flip- Flops (SR, JK & D), Counters: synchronous and asynchronous Counter. • Computer System: Comparison of Computer Organization & Architecture, Computer Components and Functions, Interconnection Structures. Bus Interconnections, Input / Output: I/O Module Programmed I/O, Interrupt Driven I/O, Direct Memory Access. <p>UNIT II:</p> <ul style="list-style-type: none"> • Memory System Organization: Classification and design parameters, Memory Hierarchy, Internal Memory: RAM, SRAM and DRAM, Interleaved and Associative Memory. Cache Memory: Design Principles, Memory mappings, Replacement Algorithms, Cache performance, Cache Coherence. Virtual Memory, External Memory: Magnetic Discs, Optical Memory, Flash Memories, RAID Levels • Instructions: Instruction Formats, Instruction Sets, Addressing Modes, Addressing Modes Examples with Assembly Language [8085/8086 CPU].

- **Processor Organization:** Structure and Function. Register Organization [8085/8086 CPU]. Basic Microprocessor operations: Data Transfer (Register / Memory) Operations, Arithmetic & Logical Operations. Instruction Cycle, Instruction Pipelining. Introduction to RISC and CISC Architecture, Instruction Level Parallelism and Superscalar Processors, Design Issues.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q.No.	Structure of the Questions	Marks
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
Total		30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Project, Role play, Creative writing, Case Study, etc.	10
	Total	20

11

REFERENCES:

1. M. Mano, Computer System Architecture 3rd edition, Pearson
2. Carl Hamacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill 2012
3. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd., 4th Edition, 2010
4. William Stallings (2010), Computer Organization and Architecture- designing for performance, 8th edition, Prentice Hall, New Jersey.
5. Anrew S. Tanenbaum (2006), Structured Computer Organization, 5th edition, Pearson Education Inc,
6. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGraw-Hill
7. Ramesh Gaonkar (2013), Microprocessor Architecture, Programming and Application with 8085, 6th edition, Penram.
8. Sarangi, S. R. (2021). Basic computer architecture, 1st edition, White Falcon Publishing.

<https://www.cse.iitd.ac.in/~srsarangi/archbooksoft.html>

9. Sarangi, Smruti R. 2023. *Next-Gen Computer Architecture*. 1st ed. White Falcon.

<https://www.cse.iitd.ac.in/~srsarangi/advbook/index.html>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Title: Fundamentals of Database Systems

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Fundamentals of Database Systems
Course Code: CHMCSI2

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction</p> <p>The Fundamentals of Database Systems course introduces students to the essential concepts of data organization, management, and retrieval. As a foundational subject in computer science and information technology, it provides the skills and knowledge needed to understand and work with modern databases, which are integral to virtually every data-driven field.</p> <p>Relevance</p> <p>In an era where data is central to decision-making and operations, efficient data management is crucial. This course equips students with the tools to understand how data is stored, structured, and accessed, making it highly relevant for both academic advancement and real-world application.</p> <p>Usefulness</p> <p>The course is particularly valuable for students aspiring to careers in software development, data analytics, systems design, and IT management. It lays the groundwork for working with databases—designing schemas, implementing queries, and ensuring data integrity and efficiency.</p> <p>Applications</p> <p>Skills learned in this course are applicable across industries such as finance, healthcare, education, business, and technology. Students will learn how to model real-life scenarios into structured data formats, create and manage databases, and develop systems for seamless information access and storage.</p> <p>Student Interest</p> <p>The practical nature of the course makes it engaging and intellectually rewarding. Students often find satisfaction in designing efficient data models, solving real-world data problems, and developing applications that rely on structured</p>

		<p>data management.</p> <p>Integration with Other Courses</p> <p>This course serves as a foundation for advanced topics in computer science. It directly supports learning in:</p> <ul style="list-style-type: none"> • Database Management Systems (DBMS) • Data Warehousing & Data Mining • Big Data Technologies • Software Engineering • Backend & Web Development • Cloud Computing <p>Demand in the Industry:</p> <p>As businesses and organizations amass ever-growing volumes of data, there is an increasing demand for professionals versed in database systems. Industries such as finance, healthcare, e-commerce, and technology actively seek individuals who can design, implement, and manage robust databases.</p> <p>Job Prospects:</p> <p>Graduates proficient in the fundamentals of database systems enjoy promising job prospects. Potential roles include database administrator, data analyst, database developer, and business intelligence analyst. These professionals play a pivotal role in ensuring the efficient and secure management of an organization's data assets.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks

<p>7</p>	<p>Course Objectives:</p> <p>CO(A) 1: To introduce the fundamental concepts of databases and DBMS architectures, including data abstraction, data independence, and data models.</p> <p>CO(A) 2: To develop the ability to design Entity-Relationship (ER) models and convert them into relational tables using proper constraints.</p> <p>CO(A) 3: To impart hands-on knowledge of SQL, including Data Definition Language (DDL), Data Manipulation Language (DML), and use of functions and subqueries.</p> <p>CO(A) 4: To explain relational data models, normalization, and database protection mechanisms for efficient and secure database design.</p> <p>CO(A) 5: To introduce the concepts of views, transaction control, and data control, focusing on data consistency, concurrency, and user management.</p>
<p>8</p>	<p>Course Outcomes:</p> <p>After successful completion of this course, students will be able to -</p> <p>CO1: Explain the architecture, components, and advantages of DBMS, and differentiate between various data models and abstraction levels.</p> <p>CO2: Design ER diagrams and convert ER models into normalized relational schemas using appropriate integrity constraints and normalization techniques.</p> <p>CO3: Write and execute SQL queries for database creation, data manipulation, and information retrieval, using joins, subqueries, and built-in functions.</p> <p>CO4: Apply relational algebra concepts and normalization principles to optimize database performance and reduce data redundancy.</p> <p>CO5: Demonstrate proficiency in database security, user access control, views, transaction management, and the effective use of AI tools for database management and query optimization.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to DBMS: Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture. • Data models: Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network). • Entity Relationship Model and ER to Table: Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization), Conceptual Design using ER (entities VS attributes, Entity Vs relationship, binary Vs ternary, constraints beyond ER) Entity to Table, Relationship to tables with and without key constraints.

- **DDL Statements:** Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables.
- **DML statements:** Viewing the structure of a table insert, update, delete, select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause.

UNIT II:

- **Relational data model:** Domains, attributes, Tuples and Relations, Relational Model Notation, Characteristics of Relations, Relational Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint.
- **Functions:** String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse) Window Functions (ROW_NUMBER(), RANK(), DENSE_RANK(), NTILE(), and OVER()).
- **Joining Tables and Subqueries:** inner join, outer join (left outer, right outer, full outer) subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries.
- **Normal forms:** Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.
- **Database Protection:** Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control, Backing Up and Restoring databases
- **Views:** Creating views with check options and without check options, altering dropping, renaming and manipulating views.
- **DCL Statements:** Creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges), User Account Locking and Expiry, Transaction control commands – Commit, Rollback, Savepoints.
- **AI Applications in Database Systems:** Introduction to AI-Assisted SQL Tools, Query Explanation and Optimization, Ethical Use and Limitations of AI in Databases, and Relevant Case Studies.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q.No.	Structure of the Questions	Marks
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Project, Role play, Creative writing, Case Study, etc.	10
		Total 20

11

REFERENCES:

1. Fundamentals of Database System, Elmasri Ramez, Navathe Shamkant, Pearson Education, Seventh edition, 2017
2. Database Management Systems, Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2014
3. Murach's MySQL, Joel Murach, 3rd Edition, 2019M. Mano, Computer System Architecture 3rd edition, Pearson.
4. Database System Concepts, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw Hill, 2017
5. MySQL: The Complete Reference, Vikram Vaswani, McGraw Hill, 2017
6. Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease, Ashwin Pajankar, BPB Publications, 2020.
7. Watt, Adrienne. *Database Design*. 2nd ed. Victoria, BC: BCcampus, 2014.
<https://opentextbc.ca/dbdesign01/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Title: Computer Science Practical 1

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Computer Science Practical 1
Course Code: CHMCSI3

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>The Major Computer Science Practical Course, which includes Digital Systems and Architecture along with Database Systems, offers a thorough, hands-on exploration of the foundational hardware and software aspects that support modern computing. This course is designed to provide students with a comprehensive understanding of digital systems, computer architecture, and effective data management within databases.</p> <p>Relevance:</p> <p>In an era where seamless integration of hardware and software is crucial, the combination of Digital Systems and Architecture with Database Systems is highly relevant. This practical course highlights the symbiotic relationship between the two, providing students with a well-rounded perspective on designing and building robust computing solutions.</p> <p>Usefulness:</p> <p>This course is highly beneficial for students looking to bridge the gap between hardware and software. By combining digital systems with database concepts, students acquire a unique skill set that empowers them to design, implement, and optimize computing systems in a holistic manner.</p> <p>Application:</p> <p>The skills gained in this practical course have direct applications in developing efficient and integrated computing solutions. Students will learn how to design digital systems, optimize hardware performance, and seamlessly integrate these systems with databases to effectively handle and manipulate data.</p> <p>Interest:</p> <p>The Major Computer Science Practical Course is designed to captivate students by offering a hands-on approach to both hardware and software components. Through practical exercises, students will design digital circuits, implement database solutions, and integrate these components, fostering a deeper</p>

		<p>understanding and appreciation of the complexities of computing systems</p> <p>Connection with Other Courses:</p> <p>This practical course serves as a nexus, linking various courses within the computer science curriculum. It establishes a strong foundation for advanced subjects like computer organization, embedded systems, software engineering, and database management. The integrated approach ensures students gain a deep understanding of the synergies between different aspects of computer science.</p> <p>Demand in the Industry:</p> <p>Professionals who can seamlessly navigate both digital systems and database management are highly sought after. Industries such as electronics, telecommunications, software development, and data analytics actively seek individuals skilled in both hardware and software, recognizing the significant practical value of this dual expertise.</p> <p>Job Prospects:</p> <p>Graduates of this practical course enjoy promising career prospects in roles that require a well-rounded understanding of computing systems. Potential job titles include systems architect, database administrator, embedded systems developer, and hardware-software integration specialist. These professionals are well-equipped to contribute to a wide range of industries in need of comprehensive computing solutions.</p>
2	Vertical 1	Major
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks

7	<p>Course Objectives:</p> <p>CO(A) 1: To enable students to simulate and verify basic digital logic circuits including logic gates, adders, subtractors, comparators, and counters using tools like Logisim.</p> <p>CO(A) 2: To develop skills in the design and implementation of combinational and sequential digital systems, such as multiplexers, flip-flops, shift registers, and ripple counters.</p> <p>CO(A) 3: To provide practical experience in relational database design, including ER modelling, normalization, and conversion to relational schemas.</p> <p>CO(A) 4: To enhance proficiency in writing and executing SQL queries, covering DDL, DML, aggregate functions, joins, subqueries, and views using MySQL or similar Database.</p> <p>CO(A) 5: To introduce the application of transaction control and data control operations such as COMMIT, ROLLBACK, GRANT, and REVOKE, for managing data integrity and security.</p>
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Design, simulate, and verify basic digital logic circuits (e.g., gates, adders, subtractors, flip-flops, counters, and multiplexers) using Logisim or equivalent tools.</p> <p>CO2: Simplify Boolean expressions and realize them as optimized digital circuits, ensuring correct logical operation through simulation.</p> <p>CO3: Design ER models and convert them into normalized relational schemas, applying up to 3rd Normal Form for structured database design.</p> <p>CO4: Execute SQL queries involving DDL, DML, joins, subqueries, and built-in functions to manipulate and retrieve data efficiently in a relational database.</p> <p>CO5: Demonstrate the use of transaction control and data control statements (COMMIT, ROLLBACK, GRANT, REVOKE) for maintaining database integrity and security.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1: Logic Gates Truth Table Verification: Study and verify the truth table of various logic gates (NOT, AND, OR, NAND, NOR, EX-OR, EX-NOR) using Logisim.</p> <p>Practical 2: Boolean Expression Simplification: Simplify given Boolean expressions and realize them using Logisim.</p> <p>Practical 3: Half/Full Adder Design: Design and verify the operation of a half/full adder using Logisim.</p> <p>Practical 4: Half/Full Subtractor Design: Design and verify the operation of a half/full subtractor using Logisim.</p> <p>Practical 5: Magnitude Comparator: Implement magnitude comparators for 1-bit, 2-bit, and 4-bit inputs using</p>

combinational logic circuits with Logisim simulation.

- Practical 6: Flip-Flop Implementation:**
Verify the operation of flip-flops (e.g., D, JK) using logic gates in Logisim.
- Practical 7: Counter Operation Verification:**
Verify the operation of a counter using Logisim.
- Practical 8: 4-Bit Shift Register Operation:**
Verify the operation of a 4-bit shift register using Logisim.
- Practical 9: Multiplexer/Demultiplexer Design:**
Design and implement expressions using multiplexers/demultiplexers in Logisim.
- Practical 10: 3-Bit Binary Ripple Counter:**
Design and implement a 3-bit binary ripple counter using JK flip-flops in Logisim.

The above practical can be performed using any open-source simulator (like Logisim) (Download it from <https://sourceforge.net/projects/circuit/>)

UNIT II:

- Practical 1: Conceptual Design Using ER Diagrams:**
Identify entities, attributes, keys, and relationships. Apply generalization and specialization.
- Practical 2: Database Management Operations:**
View all databases, create a database, view all tables in a database, create tables with and without constraints, perform CRUD operations,
- Practical 3: Table Management Operations:**
Alter a table, drop/truncate/rename tables, perform backup/restore operations on a database.
- Practical 4: Basic Queries and Aggregate Functions:**
Execute simple queries and utilize aggregate functions (e.g., COUNT, SUM, AVG).
- Practical 5: Advanced Query Functions:**
Utilize date, string, math functions and window functions such as ROW_NUMBER in queries.
- Practical 6: Join queries:**
Execute inner and outer join queries.
- Practical 7: Subqueries:**
Apply subqueries with IN and EXISTS clauses.
- Practical 8: ER Model to Relational Model Conversion and Normalization:**
Convert ER model to a relational model and apply normalization up to 3rd Normal Form.
- Practical 9: Views:**

	<p>Create views with and without check options, drop views, select data from views</p> <p>Practical 10: Data Control Language (DCL) Statements: Implement DCL statements for granting and revoking permissions, Demonstrate COMMIT, ROLLBACK statements.</p> <p>These experiments can be implemented using a database management system like MySQL.</p>																								
<p>10</p>	<p align="center">Scheme of Examination and Assessment Pattern</p> <p align="center">Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 02:00 hours</p> <p align="center">Format of Question Paper</p> <table border="1" data-bbox="558 636 1268 785"> <thead> <tr> <th>Question</th> <th>Practical Questions Based on</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td>Q.1.</td> <td>Unit – 1</td> <td>15</td> </tr> <tr> <td>Q.2.</td> <td>Unit – 2</td> <td>15</td> </tr> <tr> <td colspan="2"></td> <td align="right">Total 30</td> </tr> </tbody> </table> <p>Note:</p> <ul style="list-style-type: none"> • Certified Journal is compulsory for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination • Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process. <p>Internal Examination: Continuous Evaluation – 20 marks</p> <table border="1" data-bbox="349 1146 1421 1402"> <thead> <tr> <th></th> <th>Assessment / Evaluation</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td>1.</td> <td>Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.</td> <td>15</td> </tr> <tr> <td>2.</td> <td>Journal</td> <td>05</td> </tr> <tr> <td colspan="2"></td> <td align="right">Total 20</td> </tr> </tbody> </table>	Question	Practical Questions Based on	Marks	Q.1.	Unit – 1	15	Q.2.	Unit – 2	15			Total 30		Assessment / Evaluation	Marks	1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15	2.	Journal	05			Total 20
Question	Practical Questions Based on	Marks																							
Q.1.	Unit – 1	15																							
Q.2.	Unit – 2	15																							
		Total 30																							
	Assessment / Evaluation	Marks																							
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15																							
2.	Journal	05																							
		Total 20																							
<p>11</p>	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd., 4th Edition, 2010 2. Murach's MySQL, Joel Murach, 3rd Edition, 3rd Edition, 2019. 3. MySQL: The Complete Reference, Vikram Vaswani, McGraw Hill, 2017 4. Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease, Ashwin Pajankar, BPB Publications, 2020 5. Documentation for Logisim: https://cburch.com/logisim/docs.html 6. Documentation for MySQL: https://downloads.mysql.com/docs/refman-8.4-en.pdf 																								

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

**Vertical - 2
Minor Course**

Not Applicable

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Vertical – 3

Open Elective Course : 2 +2 Credits

**with effect from
Academic Year 2025-2026**



HSNC Board's

Smt. Chandibai Himathmal Mansukhani College

(Autonomous)

(Affiliated to the University of Mumbai)

University College Code: 217 | JD Office: T14



Estd. Year
1965

Faculty of Interdisciplinary

Vertical 3: List of Open Elective Skill Based Courses for First Year: Semester - I

Sr. No.	Nomenclature of the Paper
1	Basic Computer Skills for Digital Age
2	Visual Design and Digital Tools : A Foundation For Animation
3	Basic Tools of AI for Economics and Education
4	Communicative English
5	Urbanization and Real Estate: Concepts and Contemporary Scenarios
6	Business of Travel and Tours
7	Managing Family Wealth Through Family Office
8	Web Designing Essentials: HTMLI and CSS Styling Techniques
9	Basics of Nutrition
10	Lessons of Reel Making
11	Performing Arts
12	Data Analysis with Excel
13	Political Communication and Media Skills
14	Stress Management-I
15	Social Media and Communication
16	Mushroom Cultivation: Training and Trading
17	Yoga and Fitness
18	Basic Perfumeries Course (Level-I)
19	Soft Skills for Corporate Readiness
20	Beautician : Strategic Business Planning
21	Current Trends of Fashion Design: Financial Perspective
22	Basics of Accounting-I
23	Digital Marketing
24	Online Trading in Stock Market
25	Event Management Course in Sindhi



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Vertical – 4

Vocational Skill Course (VSC) : 2 Credits
Skill Enhancement Course (SEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Title: Introduction to Programming with Python

**Vertical - 4
Major Subject - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Introduction to Programming with Python

Course Code: CHMCSI5

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction: The Introduction to Programming with Python course is designed as a starting point for anyone new to coding, offering a solid foundation in one of the most versatile and accessible programming languages. Known for its clear syntax and ease of learning, Python is an excellent choice for beginners looking to develop essential programming skills.</p> <p>Relevance: In today's technology-driven world, programming has become a vital skill across numerous fields. Python, as a high-level interpreted language, holds particular relevance due to its wide range of applications—including web development, data analysis, artificial intelligence, and automation—making it a valuable tool for professionals in virtually any domain.</p> <p>Usefulness: The course offers a comprehensive introduction to core Python concepts, including syntax, data structures, and control flow. It enables learners to develop the skills needed to write effective and efficient code. Given Python's versatility, the knowledge gained is applicable across a broad spectrum of programming challenges and practical applications.</p> <p>Application: By the end of the course, participants will be equipped to use Python for practical purposes such as automating tasks, solving everyday problems, and building simple programs. This foundational knowledge also paves the way for further study in specialized fields like data science, machine learning, or web development.</p> <p>Interest: Python's easy-to-understand syntax and powerful libraries make it an ideal language for beginners. The course is designed to engage learners with hands-on projects, inspiring a passion for coding and problem-solving.</p> <p>Connection with Other Courses: Python serves as an ideal gateway into the broader world of programming, thanks to its compatibility with other languages and technologies. The foundational skills developed in this course prepare learners to confidently progress to more</p>

		<p>advanced programming topics and specialized fields such as data science, machine learning, web development, and beyond.</p> <p>Demand in the Industry: Python continues to rise in popularity across industries due to its versatility, clean syntax, and strong community support. Its widespread adoption in fields such as technology, finance, healthcare, and academia has created a strong demand for professionals skilled in Python, making it a valuable asset for career advancement in today's job market.</p> <p>Job Prospects: Completing this course will open up entry-level opportunities in software development, quality assurance, data analysis, and scripting. Python developers are in high demand for their ability to quickly create prototypes and contribute across various stages of the software development process.</p>
2	Vertical 1	VSC
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: To introduce the fundamental concepts of Python programming, including syntax, variables, data types, operators, and basic input/output operations.</p> <p>CO(A) 2: To develop control flow logic using conditional and iterative statements, enhancing problem-solving and algorithmic thinking.</p> <p>CO(A) 3: To enable students to implement core data structures like lists, tuples, dictionaries, arrays, and strings, and perform manipulations using built-in functions.</p> <p>CO(A) 4: To impart functional and modular programming skills, including the use of user-defined functions, recursion, lambda expressions, modules, and packages.</p> <p>CO(A) 5: To expose students to file handling, regular expressions, and date-time operations, providing a solid foundation for real-world programming and data processing applications.</p>	

8	<p>Course Outcomes:</p> <p>CO1: Understand and apply the syntax and semantics of Python programming language, including variables, operators, and basic input/output.</p> <p>CO2: Implement control flow using decision-making and looping constructs such as if, for, while, break, and continue statements</p> <p>CO3: Manipulate and process structured data using Python data structures like arrays (NumPy), strings, lists, tuples, and dictionaries.</p> <p>CO4: Design and develop modular programs using functions and modules including recursion, parameter handling, and AI-assisted coding tools.</p> <p>CO5: Handle external data through file operations, regular expressions, and date-time modules for solving real-time programming problems.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1: Overview and Basic Elements of Python Programming:</p> <ol style="list-style-type: none"> a) Utilizing IDLE as a basic calculator. b) Practicing print statements using IDLE. c) Programs that create and use variables. d) Developing programs with input and output operations. e) Implementing programs that explore data types and type conversions. f) Creating programs that use command line arguments. g) Writing programs to understand operator precedence and associativity. <p>Practical 2: Control Statements:</p> <ol style="list-style-type: none"> a) Writing programs using if, if...else, and if...elif...else statements. b) Developing programs with nested if statements. c) Implementing decision-making using the match...case statement. d) Writing programs using while loops. e) Creating iterative solutions with for loops. f) Utilizing jump statements such as break and continue for flow control. g) Understanding the use of return for outputting function results and assert for debugging by validating conditions. <p>Practical 3: Arrays:</p> <ol style="list-style-type: none"> a) Implementation of array creation techniques, including indexing and slicing operations. b) Development of programs to perform fundamental and mathematical operations on arrays. c) Utilization of NumPy arrays with emphasis on slicing, conventional indexing, and advanced indexing methodologies. d) Examination of array properties, including dimensionality and other structural attributes.

Practical 4:

Functions:

- a) Development of programs utilizing user-defined functions.
- b) Implementation of various types of function parameters.
- c) Construction of recursive function-based solutions.
- d) Exploration and application of built-in functions.
- e) Use of lambda (anonymous) functions in program logic.
- f) Composition of functions to build complex operations.
- g) Integration of standard library modules and creation of custom modules.

Practical 5:

Strings:

- a) Development of programs for string creation and manipulation.
- b) Implementation of indexing and slicing techniques for string processing.
- c) Utilization of built-in string functions for various operations.
- d) Construction of programs focused on string formatting.
- e) Design of programs to insert substrings into existing strings.

UNIT II:

Practical 6:

Exploring List, Tuples:

- a) Development of programs to create and initialize lists.
- b) Implementation of programs utilizing list functions and methods.
- c) Construction of programs that perform basic list operations.
- d) Exploration of list slicing techniques and working with nested lists.
- e) Design of programs using list comprehension for efficient processing.
- f) Creation and initialization of tuples through programming.
- g) Utilization of built-in functions for manipulating tuples.

Practical 7:

Working with Dictionaries:

- a) Development of programs to create dictionaries and perform various operations on them.
- b) Construction of programs to iterate over dictionaries using loops.
- c) Implementation of programs utilizing different operators on dictionaries.
- d) Exploration of various built-in dictionary methods through programming.
- e) Implementation of programs to create dictionaries using dictionary comprehension.

Practical 8:

Files in Python:

- a) Development of programs to create and open text and binary files for reading and writing data.
- b) Implementation of programs utilizing the tell() and seek() methods for file manipulation.
- c) Design of programs to serialize and deserialize Python objects using the pickle module.
- d) Construction of programs to perform operations with directories.

- Practical 9: Regular Expressions:**
- Development of programs to search for specific patterns within strings using sequence and special characters.
 - Implementation of programs to substitute one pattern with another in a string.
 - Design of programs that apply regular expressions for pattern matching in files.
- Practical 10: Date And Time in Python:**
- Development of programs to display the current time, date, and date-time information.
 - Implementation of programs to create, combine, and format dates and times.
 - Construction of programs to find, compare, and sort dates in a list.
 - Design of programs to measure the execution time of code.
 - Exploration of programs utilizing the calendar module for various operations.

Supplementary Exercise: Demonstration of GitHub Copilot AI tool for programming assistance, including code suggestion, auto-completion, and debugging support.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – 1	15
Q.2.	Unit – 2	15
		Total 30

Note:

- Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15

	2. Journal	05
	Total 20	
11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries, Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014 2. Programming through Python, M. T Savaliya, R. K. Maurya & G M Magar, Sybgen Learning India, 2020. 3. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 4. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017 5. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018 6. Python Programming: Using Problem Solving Approach, ReemaThareja, Oxford Univeristy Press, 2017 7. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019 8. Documentation for Python https://docs.python.org/3.13/tutorial/index.html 	

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Title: LINUX Operating System

Vertical - 4

Skill Enhancement Course (SEC) - 2 Credits

**with effect from
Academic Year 2025-2026**

Title: LINUX Operating System

Course Code: CHMCSI6

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction: The Linux Operating System course provides a foundational yet technically rigorous introduction to computing, focusing on the architecture, functionality, and practical use of this widely adopted open-source operating system.</p> <p>Relevance: Linux plays a critical role across multiple industries—including server administration, software development, cybersecurity, cloud computing, and the Internet of Things—making this course highly pertinent to the demands of today's digital ecosystem.</p> <p>Usefulness: Linux is the dominant operating system in global server environments, making proficiency in it essential for effective server management and maintenance. A significant number of development tools and environments are built for or optimized on Linux, greatly enhancing a developer's efficiency and capabilities. Renowned for its strong security architecture, Linux is a cornerstone in the cybersecurity domain, where in-depth knowledge is crucial for professionals. Major cloud platforms are built on or heavily rely on Linux, making it a vital skill for cloud administrators. Additionally, Linux's widespread use in IoT devices and embedded systems highlights its importance for professionals working in these rapidly growing technological areas.</p> <p>Application: This course introduces students to the fundamental concepts and practical uses of Linux, with focused coverage of key domains including server administration, software development, cybersecurity, cloud computing, and the Internet of Things (IoT).</p> <p>Interest: Thanks to its open-source nature and wide-ranging capabilities, Linux appeals to individuals who value powerful command-line tools, as well as those seeking a stable, reliable operating system with a robust command-line interface.</p> <p>Connection with Other Courses: The course complements network administration curricula by</p>

		<p>incorporating essential Linux commands, while also aligning with software development courses to promote a well-rounded understanding of modern computing environments.</p> <p>Demand in the Industry: Industry leaders consistently value Linux for its stability, security, and cost-efficiency, driving strong and ongoing demand for professionals skilled in Linux systems.</p> <p>Job Prospects: Graduates of the Linux Operating System course are equipped to pursue a range of careers, including positions as system administrators, network administrators, DevOps engineers, cloud administrators, cybersecurity analysts, and software developers.</p>
2	Vertical 1	SEC (Skill Enhancement Course)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: To introduce the fundamental concepts, architecture, and features of the Linux operating system.</p> <p>CO(A) 2: To enable learners to use essential Linux shell commands, including regular expressions, for basic system operations.</p> <p>CO(A) 3: To provide knowledge on setting up environment variables and managing file permissions for secure Linux system operations.</p> <p>CO(A) 4: To develop proficiency in using Linux text editors and writing shell scripts for automating routine administrative tasks.</p> <p>CO(A) 5: To familiarize learners with installing compilers and developing basic programs using Python in the Linux environment.</p>	

8	<p>Course Outcomes:</p> <p>CO1: To explain the fundamental concepts, architecture, and features of the Linux operating system.</p> <p>CO2: To apply various Linux shell commands, including regular expressions, for effective system management.</p> <p>CO3: To configure Linux environment variables and manage file permissions to ensure system security.</p> <p>CO4: To develop shell scripts using Linux text editors for automating system tasks.</p> <p>CO5: To install compilers and write basic programs in Python on the Linux platform.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1:</p> <p>a) Introduction to Linux Operating System and Basics: History of Linux, GNU Info and Utilities, Various Linux Distributions, The Unix/Linux architecture, Features of Unix/Linux</p> <ul style="list-style-type: none"> • Explore the role of an operating system and its fundamental functions. • Examine the history of UNIX and the evolution of Linux from it. • Illustrate and explain the basic architecture of Linux, including User Space, Kernel, and Hardware. • Identify five key features that set Linux apart from other operating systems. <p>b) Installation of Ubuntu Linux Operating System: Booting and Installing from USB/DVD, Using Ubuntu Software Center / Using Synaptic, Exploring useful software packages.</p> <ul style="list-style-type: none"> • Demonstration or video tutorial on installing the Linux Operating System. • Log into Ubuntu and execute the command 'sudo apt update && sudo apt upgrade.' • Install and explore the Ubuntu Software Center and Synaptic Package Manager. <p>Practical 2: Becoming an Ubuntu Power User: Administering system and user settings, Learning Unity keyboard shortcuts, Using the Terminal</p> <ul style="list-style-type: none"> • Learn user management commands in Linux. • Execute user management commands such as who, whoami, su, sudo, login, logout, exit, passwd, useradd/adduser, usermod, userdel, groupadd, groupmod, groupdel, gpasswd, chown, chage, chgrp, chfn, etc. • Practice Unity/GNOME shortcuts like Ctrl+Alt+T and Alt+Tab. <p>Practical 3:</p> <p>a) Linux Basics: Starting the shell, Shell prompt, Command structure, File Systems and Directory Structure, man pages, more documentation pages</p> <ul style="list-style-type: none"> • Learn general-purpose Linux commands and programming within the Linux editor environment. • Explore the Linux filesystem hierarchy (tree structure) with a diagram of the file

system.

- Execute commands like 'pwd', 'cd /', and 'ls -l' to observe the structure.
- Install the 'tree' command with 'sudo apt install tree' and explore the /etc directory.
- Use 'man ls' and 'info bash' to access and read the documentation.

b) File System Commands:

touch, help, man, more, less, pwd, cd, mkdir, rmdir, ls, find, etc.

- Understand file system management commands in Linux.
- Execute file system commands such as touch, help, man, more, less, pwd, cd, ls, find, and others.
- Use file and directory management commands like mkdir, rmdir, echo, sort, head, tail, less, more, file, type, split, vim, and more.

Practical 4: File Handling Commands:

cat, cp, rm, mv, more, file, wc, od, cmp, diff, comm, gzip, gunzip, zip, unzip, tar, ln, umask, etc.

- Understand file handling commands in Linux.
- Execute file handling commands such as cat, cp, rm, mv, more, file, wc, od, cmp, diff, comm, gzip, gunzip, zip, unzip, tar, ln, umask, bzip2, locate, and more.
- Experiment with the 'ln' command for creating links and 'umask' to manage default permission settings.

Practical 5:

a) General Purpose Utility Commands:

cal, date, echo, man, printf, passwd, script, who, uname, tty, stty, etc.

- Understand general-purpose utility commands in Linux.
- Execute general-purpose utility commands such as cal, date, echo, man, printf, passwd, script, who, uname, tty, stty, and others.

b) Linux File Permissions: Understanding Linux file permissions, Using Linux groups. Decoding file permissions, Changing security settings, chmod, chown, chgrp.

- Understand file permission commands in Linux.
- Study file and directory permissions in both single and multi-user environments.
- Use 'ls -l' to view file permissions.
- Modify permissions with commands like 'chmod', 'chown', and 'chgrp'.
- Explain the meaning of permissions such as 755, 644, and others.

UNIT II:

Practical 6: Linux Security:

Understanding Linux Security, working with passwords, Understanding ssh

- Switch to the root user using 'sudo su' and observe the associated privileges.
- Change a user's password with the command 'sudo passwd username'.
- Install and enable SSH using 'sudo apt install openssh-server'.

- Start the SSH service and test a local SSH connection.

Practical 7:

a) Networking Commands:

who, whoami, ping, telnet, ftp, ssh, etc.

- Verify network connectivity and examine active sessions.
- Run the following commands: who, whoami, ping google.com, ssh user@localhost, ip a, ss -tuln.

b) Editors:

vi, sed, awk

- Open a file in vi, insert text, and save the changes.
- Use sed to replace, insert, or delete lines within a file.
- Write awk scripts to print specific patterns, manipulate columns, and perform arithmetic operations.

c) Simple Filters and I/O Redirection:

cut, paste, grep family, tee, uniq, tr, piping and redirection etc.

- Run filter and redirection commands such as head, tail, cut, paste, sort, grep, tee, uniq, and tr on sample text files.
- Combine filters using pipes and redirect the output to files.

Practical 8:

Shell Scripting:

Defining variables, reading user input, exit and exit status commands, expr, test, [], if conditional, logical operators, Conditions (for loop, until loop, and while loop), arithmetic operations, Redirecting input/output in scripts, creating your own redirection.

- Write scripts to perform arithmetic calculations, calculate simple and compound interest, determine if a year is a leap year, max/min of three numbers, compare two strings, etc.
- Utilize loops (for, while, until) and conditional statements (if, case).
- Develop a script for a menu-driven calculator using case statement.
- Implement the 'read' command and check for the existence of files or directories.

Practical 9:

Working and Managing Processes:

sh, ps, kill, nice, at, batch, etc.

- Execute process management commands such as ps, pstree, top, nice, kill, pkill, killall, xkill, fg, bg, pgrep, renice, etc.
- Start a process with a low priority and modify its priority using renice.

Practical 10:a) **Job scheduling commands:**

ps, nice, renice, at, batch, cron table

- Schedule a one-time task using the 'at' command.
- View scheduled 'at' jobs with 'atq' and remove them using 'atrm'.
- Set up a cron job to execute a script daily or weekly.

b) Installation of Python Compiler and Environment Setup and Basic programming using Python.

10**Scheme of Examination and Assessment Pattern**

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – 1	15
Q.2.	Unit – 2	15
		Total 30

Note:

- **Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11**REFERENCES:**

1. Linux Command line and Shell Scripting Bible, Richard Blum, Wiley India.
2. Unix: Concepts and Applications, Sumitabha Das, 4th Edition, McGraw Hill.
3. Official Ubuntu Book, Matthew Helmke& Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, 8th Ed.
4. Linux Administration: A Beginner's Guide, Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008.
5. Linux: Complete Reference, Richard Petersen, 6th Edition, Tata McGraw-Hill
6. Beginning Linux Programming, Neil Mathew, 4th Edition, Wiley Publishing, 2008.
7. <https://ubuntu.com/tutorials/command-line-for-beginners#1-overview>
8. <https://oracle-samples.github.io/oltrain/tracks/ol/intro-ol/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- I

Vertical – 5

Ability Enhancement Course (AEC) : 2 Credits

Value Education Course (VEC) : 2 Credits

Indian Knowledge System (IKS) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year B.Sc.

Semester- I

Vertical – 5

**Ability Enhancement Course (English)
2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Introduction to Communication Skills in English
Course Code: CHMBSCAECI

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Effective communication is the cornerstone of academic and professional success. This course introduces learners to foundational skills in English communication, with a focus on both oral and written competencies essential in academic, social, and workplace contexts. It aims to equip learners with the ability to read critically, write precisely, speak confidently, and listen actively. Emphasis is placed on building clarity, coherence, and conciseness in communication, along with an understanding of audience, purpose, and tone.</p> <p>The course integrates grammar reinforcement, vocabulary building, reading comprehension, and practice-oriented modules such as email etiquette, group discussion, and formal writing. Through dynamic classroom interactions and practical assessments, learners will gain confidence in using English effectively and purposefully.</p>
2	Vertical 4	Ability Enhancement Course
3	Type Teaching Methods:	Theory+ Practicum (Lecture/ Discussion/ Presentation/ Reading sessions/ Worksheets/ etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A)1: To introduce learners to the fundamentals of effective communication and its components.</p> <p>CO(A)2: To enhance learners' reading comprehension through exposure to multiple genres and contexts.</p> <p>CO(A)3: To develop grammatical accuracy and lexical resourcefulness for academic and professional communication.</p> <p>CO(A)4: To strengthen verbal and non-verbal presentation skills and promote interactive speaking abilities.</p>	

	CO(A)5: To build competence in real-world writing tasks such as email drafting, bio-data preparation, and descriptive writing.
8	Course Outcomes: Student will be able to CO-1: Understand and apply key principles of effective communication in varied contexts. CO-2: Comprehend and analyze written texts using appropriate reading strategies. CO-3: Recognize and correct common grammatical and lexical errors. CO-4: Engage in clear, confident, and context-appropriate spoken interactions. CO-5: Produce structured, coherent, and grammatically correct written content for academic and workplace use.

Syllabus

UNIT I: Foundations of English Communication

A) Core Concepts of Communication

- Principles of Effective Communication: The 7 Cs
- Verbal and Non-verbal Communication with Examples
- Cross-cultural Communication in the Globalized World
- Technology in Communication: Email, Messaging, Video Conferencing
- Listening for Detail and Intent: Barriers to Listening and Strategies

B) Reading Comprehension

- Understanding the Main Idea and Supporting Details
- Interpreting Tone, Purpose, and Bias
- Using Context Clues for Vocabulary Building
- Reading Visual Texts: Graphs, Charts, and Infographics
Sample readings will include excerpts from news articles, reports, editorials, and educational essays (200–250 words).

C) Grammar and Vocabulary

- Subject-Verb Agreement
- Sentence Structures
- Punctuation and Capitalization
- Commonly Confused Words
- Editing and Proofreading Practice

A remedial and functional approach will be followed with contextual exercises.

UNIT II: Applied Communication Skills

A) Speaking and Listening Skills

- Introducing Oneself in Academic/Professional Settings
- Participating in Group Discussions and Expressing Opinions
- Delivering a Short Speech (2–3 minutes) on Familiar Topics
- Understanding and Responding to Instructions
- Listening Comprehension Practice through Audio/Video Clips

B) Functional Writing Skills

- Formal Email Writing with Subject and Tone Sensitivity
- Descriptive Paragraph Writing (People, Places, Processes)
- Bio-data and Resume Writing
- Drafting Job Applications (Solicited and Unsolicited)
- Writing a Statement of Purpose

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

Question No	Nature of Questions	Marks
Q. 1	Short Notes (Attempt any 3 out of 5) - Unit 1 OR Essay-Type Question (Attempt any 1 out of 2)- Unit 1	15
Q. 2	Short Notes (Attempt any 3 out of 5) - Unit 2 OR Essay-Type Question (Attempt any 1 out of 2)- Unit 2	15
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Students are required to draft a job application letter along with a resume using the following AI assistance: Canva Resume Builder, Resume.oj, Zety, Novopresume, Rezi etc OR Draft an SoP with the help of the following AI assistance: Quillbot, Yocket, Writesonic, Jasper AI	15
2.		05
	Total	20


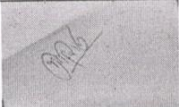
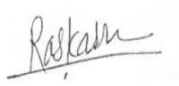
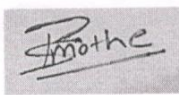
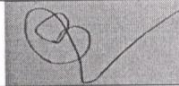
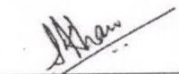
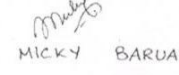

11

REFERENCES:

1. Adler, Ronald B., et al. *Understanding Human Communication*. 15th ed., Oxford UP, 2021.
2. Bailey, Stephen. *Academic Writing: A Handbook for International Students*. 5th ed., Routledge, 2018.
3. Barrass, Robert. *Students Must Write: A Guide to Better Writing in Coursework and Examinations*. Routledge, 2005.
4. Brown, Gillian, and George Yule. *Teaching the Spoken Language: An Approach Based on the Analysis of Conversational English*. Cambridge UP, 1983.
5. Carnegie, Dale. *The Quick and Easy Way to Effective Speaking*. Pocket Books, 2006.
6. Chaney, Lillian, and Jeanette Martin. *Intercultural Business Communication*. 6th ed., Pearson, 2014.
7. Cullen, Pauline, et al. *English Grammar in Use Supplementary Exercises*. Cambridge UP, 2019.
8. Eastwood, John. *Oxford Guide to English Grammar*. Oxford UP, 2005.

9. Gerson, Sharon J., and Steven M. Gerson. *Technical Communication: Process and Product*. 9th ed., Pearson, 2021.
10. Hewings, Martin. *Advanced Grammar in Use: A Self-study Reference and Practice Book for Advanced Learners of English*. 3rd ed., Cambridge UP, 2013.
11. Jones, Leo. *Functions of English: Communication Practice in English*. Cambridge UP, 1981.
12. Kumar, Sanjay, and Pushp Lata. *Communication Skills*. 2nd ed., Oxford UP, 2018.
13. Lynch, Tony. *Listening in Language Learning*. Longman, 1988.
14. McCarthy, Michael, and Felicity O'Dell. *Academic Vocabulary in Use*. Cambridge UP, 2008.
15. Nordquist, Richard. *The Essentials of English Grammar*. McGraw-Hill Education, 2016.
16. Quirk, Randolph, et al. *A Comprehensive Grammar of the English Language*. Longman, 1985.
17. Seely, John. *The Oxford Guide to Writing and Speaking*. Oxford UP, 2005.
18. Straus, Jane, et al. *The Blue Book of Grammar and Punctuation*. 12th ed., Jossey-Bass, 2021.
19. Wallace, Catherine. *Reading*. Oxford UP, 1992.
20. Zinsser, William. *On Writing Well: The Classic Guide to Writing Nonfiction*. Harper Perennial, 2016.

Syllabus Committee:

Sr. No	Name of the Faculty	Designation and College	Signature
1.	Prof. (Dr.) Kailas Aute	Professor & Head, Dept. of English, Smt. CHM College	
2.	Prof. (Dr.) B. R. Hiramani,	(VC Nominee, University of Mumbai) Pancham Khemraj College, Sawantwadi	
3.	Prof. (Dr.) Vikas Raskar	(Subject Expert outside University) Hutatma Rajguru Mahavidyalay, Rajguru Nagar, Khed, (Affiliated to Savitribai Phule University)	
4.	Prof. (Dr.) Prashant Mothe	(Subject Expert outside University) Aadarsh Mahavidyalay, Umerga, Dharashiv, (Affiliated to Dr. Baba Saheb Ambedkar Marathwada University)	
5.	Mr. Ananda Pandhare	Asst. Professor, Dept. of English, Smt. CHM College	
6.	Ms. Sana Khan	Asst. Professor, Dept. of English, Smt. CHM College	
7.	Dr. Micky Barua	Faculty Vidyalkar Institute of technology, Alumni Member	 MICKY BARUA
8.	Ms. Sofy Verghese	Accenture, Industry Representative	

Name & Signature of the Ad-hoc BoS Chairperson: Prof. (Dr). Kailas Aute



Name & Signature of the Dean: Prof. (Dr). Nitin Arekar



Smt
Smt. Chandibai Himathmal Mansukhani College
(Autonomous)

First Year

Semester- I

Title: Communication Skills in Sindhi

Vertical - 5
Ability Enhancement Course
2 Credit

with effect from
Academic Year 2025-2026

Title: Communication Skills in Sindhi
COURSE CODE: CHMSINIAEC

Sr. No.	Heading	Particulars
1	Description the Course:	Communication is the core component of commerce and trade. In communication, language plays very significant role. If a student has mastered the skills of language, undoubtedly, he or she would be able to communicate in the best manner. In this course basic part of Sindhi language would be taught based on the NEP 2020. Innovative approaches like critical thinking, creative mind, and use of technology will lead to communicating and participating with different groups. The vocabulary section would be given prominence. The course would be in the Devanagari script so that it can attract majority of the students. Even non-Sindhi students shall have opportunity to adopt this course.
2	Vertical 1	AEC – Ability Enhancement Course
3	Type	Theory + Practicum (Teaching Method: Lecture/ Discussion/Reading)
4	Credit	2 credits (1 credit = 15 hours for theory or 30 hours of Practical work in a semester)
5	Hours allotted	30 Hours
6	Marks allotted	50Marks
7	Course Objectives: After successful completion of this course: CO(A) 1: The learner will get understanding of communication skills. CO(A) 2: The learner will understand how to accurate the pronunciation of special words in Sindhi CO(A) 3: The learner will improve the conversation skill in Sindhi. CO(A) 4: The learner will become best communicator in Sindhi language	
8	Course Outcomes: Student will be able to CO1: Know the basic special features of Sindhi language. CO2: Understand communication skills. CO3: Knowing the conversation with businessmen and customers CO4: Know the etiquettes with parents, relatives, friends and others in effective way	

Syllabus

UNIT I: Fundamental of Sindhi Communication

- Introduction of Communication skills through Pictorial Presentation
- Importance of Language
- Basic aspects of language:
 - i) Types of Language, ii) Role of Language, iii) Changes in Language iv) Non-violent aspects of language v) Language & New generation vi) Language & Modern technology
- New Education Policy (NEP) & Importance of language
- Sindhi language: (Special features of spoken Sindhi language with pronunciation through audio visual presentation)

UNIT II: Functional Communication

- Importance of Communication
- Types of Communication (Presentation through video clips):
 - i) Verbal, ii) non-verbal, iii) Written iv) Digital Communication
- Characteristics of Communication
- Obstacles in Communication of Sindhi Language
- Methods of Best Communication through role plays
- Spoken Sindhi in Business
- Conversation with customers and proprietors

Syllabus

यूनित १: सिंधी राबते जो बुनियाद

- संचारी भाषा: वाक्फियत
- भाषा जी अहमियत
- भाषा जा बुनियादी पहलू
१) किस्म, २) भाषा जो किरदार, ३) भाषा मे तबदीलियूं, ४) भाषा जा अहिसासाती पहलू ५) बोली ऐ नई पीड़ी
६) बोली ऐ जदीद टेकनालाजी
- नई तैलीमी नीति ऐं बोलियुन जी अहमियत
- असां जी सिंधी बोली

यूनित २: अमली राबतो (असराइतो गालाइण जो तरीको)

- राबते जी अहमियत
- राबते जा किस्म
१) जिबानी राबतो, २) गैर जिबानी राबतो, ३) लिख्त राबतो ४) डिजीटल राबतो राबते मां फायदा
- राबते में रंडकुं
- बेहतर राबते जा तरीका
- ग्राहकन सां सिंधी बोलीअ मे गुफ्तगू
- कारोबार में सिंधी गालाइण

10**Scheme of Examination and Assessment Pattern**

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

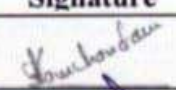
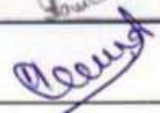
Question No	Nature of Questions	Marks
Q1.A)	Objective Type Questions (Unit- I)	05
Q1.B)	Attempt ANY 2 out of 4- (5 marks each) (Unit-I)	10
Q2.A)	Objective Type Questions (Unit- II)	05
Q2.B)	Attempt ANY 2 out of 4- (5 marks each) (Unit-II)	10
Total		30
Internal Examination: Continuous Evolution - 20 marks		Total 30

	Assessment / evaluation	Marks
1.	Written assignment on any one of the following topics 1) Draft a Notice and Report writing in Sindhi 2) Simulate dialogues such as interview, daily conversation and public speaking in Sindhi (Students are required to use AI assistance in the preparation of their drafts. Eg: Microsoft Copilot, Google Gemini, Google voice Typing tool)	15
2.	Class Attendance and Participation	05
Total		20

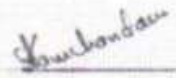
11**REFERENCE BOOKS:**

1. Sanchari Basha – By Dr. Pushpa Kodwani
2. Sindhi Pahakaa – Dr. Jetly M.K.
3. Sindhi Muhavahra – By Hardwani Lachhman
4. Sindhi Adhyat mak Shabdhkesh – By Hardwani Lachhman
5. Acho Sindhi Sikhu – By Hardwani Lachhman


Syllabus Committee:

Sr No	Name of the Faculty	Designation and College	Signature
1.	Dr. Kajal Ramchandani	H.O.D. of Jai Hind College	
2.	Mrs. Komal Totani	Faculty In-Charge, Smt. CHM College	

Name & Signature of the BoS Chairperson: (Dr. Kajal Ramchandani)



Name & Signature of the Dean: (Dr. Nitin Arekar)





**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester - I

**Title: Environmental Management and
Sustainable Development-I**

**Vertical - 5
VEC Subject - 2 Credits**

**With effect from
Academic Year 2025-2026**

Title: Environmental Management and Sustainable Development-I
Course Code: CHMVECI

Sr. No.	Heading	Particulars
1	Description the Course:	This course introduces students to the basics of environmental management and sustainable development. It explains how ecosystems work, the importance of biodiversity, and the need to protect our natural resources. Students will learn about different environmental problems, human impact on nature, and how to manage disasters. The course also covers Indian environmental movements, ethics, and the role of public awareness. Real-life examples and case studies help students understand the connection between nature and human communities in a simple and practical way.
2	Vertical 5	VEC
3	Type & Teaching Methods	Theory + Practicum Lectures/Discussions/Presentations/Case Studies, etc.
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<p>CO(A)1: To introduce about ecosystems, biodiversity and to make aware for the need of conservation.</p> <p>CO(A)2: To sensitize students towards environmental concerns, issues, and impacts of human population.</p> <p>CO(A)3: To analyze the impact of human population growth and development activities on the environment, including issues related to displacement, disaster response, and rehabilitation.</p> <p>CO(A)4: To foster awareness of environmental ethics and the role of cultural and social movements in shaping sustainable environmental practices through communication, policy, and activism.</p>
8	Course Outcomes:	<p>Student will be able to</p> <p>CO1: Explain the interrelationships within ecosystems and analyze energy flow and succession, using examples from various ecological zones.</p> <p>CO2: Critically evaluate biodiversity levels and conservation strategies, applying knowledge of endemic species, threats, and ecological services to real-world scenarios.</p> <p>CO3: Assess the socio-environmental implications of population growth, displacement, and disasters, incorporating case studies to understand sustainable development challenges.</p> <p>CO4: Demonstrate an understanding of environmental ethics and advocacy, by interpreting the influence of cultural values, environmental movements, and communication strategies on sustainability.</p>

9

Syllabus

UNIT I: Ecosystems, Biodiversity and Conservation

- Introduction, structure, and function of ecosystems; Energy flow: food chains, food webs and ecological succession. Case studies of the following:
 - Forest ecosystem
 - Grassland ecosystem
 - Desert ecosystem
 - Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)
- Levels of biological diversity: genetic, species and ecosystem diversity; Biogeographic zones of India; Biodiversity patterns.
- India as a mega-biodiversity nation; Endangered and endemic species of India.
- Threats to biodiversity: Habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.
- Ecosystem and biodiversity services: Ecological, economic, social, ethical, aesthetic and informational value.

UNIT II: Human Communities and the Environment

- Human population growth: Impacts on environment, human health and welfare.
- Resettlement and rehabilitation of project affected persons; case studies.
- Disaster management: floods, earthquake, cyclones and landslides.
- Environmental movements: Chipko, Silent valley, Bishnois of Rajasthan.
- Environmental ethics: Role of Indian and other religions and cultures in environmental conservation.
- Environmental communication and public awareness, case studies (e.g. CNG vehicles in Delhi).

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hours

Format of Question Paper

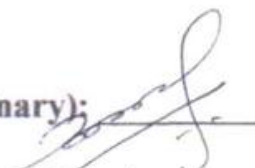
Attempt any 3 out of 4 questions.

Question No	Nature of Questions	Marks
Q1	Theory based on Unit I	10
Q2	Theory based on Unit I	10
Q3	Theory based on Unit II	10
Q4	Theory based on Unit II	10
TOTAL		30

Internal Examination: Continuous Evaluation - 20 marks		
	Assessment / evaluation	Marks
1.	Class Test, Creative writing/visits/role play (Short notes/ MCQ's/ Match the Pairs/ Answer in one sentence/ Quiz)	10
2.	Project /Presentation / Viva/Group Discussion/Case study	10
TOTAL		20

11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Carson, R. (2002). <i>Silent Spring</i>. Houghton Mifflin Harcourt. 2. Gadgil, M., & Guha, R. (1993). <i>This Fissured Land: An Ecological History of India</i>. University of California Press. 3. Gleeson, B., & Low, N. (Eds.). (1999). <i>Global Ethics and Environment</i>. Routledge. 4. Gleick, P. H. (1993). <i>Water in Crisis</i>. Pacific Institute for Studies in Development, Environment & Security; Stockholm Environment Institute; Oxford University Press. 5. Sodhi, N. S., Gibson, L., & Raven, P. H. (Eds.). (2013). <i>Conservation Biology: Voices from the Tropics</i>. John Wiley & Sons. 6. Thapar, V. (1998). <i>Land of the Tiger: A Natural History of the Indian Subcontinent</i>. 7. Warren, C. E. (1971). <i>Biology and Water Pollution Control</i>. W. B. Saunders. 8. Wilson, E. O. (2006). <i>The Creation: An Appeal to Save Life on Earth</i>. W. W. Norton. 9. Harper, Charles L. (2017). <i>Environment and Society: Human Perspectives on Environmental Issues</i> (6th Edition). Routledge. 10. Rajagopalan, R. (2011). <i>Environmental Studies: From Crisis to Cure</i>. Oxford University Press. 11. Harris, Frances (2012). <i>Global Environmental Issues</i> (2nd Edition). Wiley-Blackwell.
-----------	--

Name & Signature of the Dean & Ad-hoc BoS Chairperson (Interdisciplinary):


 Dr. Nitin Arekar



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester - I

Title: Indian Knowledge System

**Vertical - 5
IKS Subject - 2 Credits**

**With effect from
Academic Year 2025-2026**

Title: Indian Knowledge System
Course Code: CHMIKSI

Sr. No.	Heading	Particulars
1	Description the Course:	This course introduces students to the Indian Knowledge System (IKS), emphasizing its historical depth, cultural relevance, and interdisciplinary value. Rooted in the context of Indian civilization, it explores the holistic development of knowledge from ancient to pre-modern times, including contributions in medicine, mathematics, logic, linguistics, governance, arts, and sciences. By revisiting the traditional education systems and intellectual heritage of India, the course encourages learners to connect ancient insights with contemporary disciplines. It aims to enhance awareness, foster appreciation of indigenous wisdom, and reveal the interconnectedness of various streams of knowledge, aligning with the goals of the NEP 2020.
2	Vertical 5	IKS
3	Type & Teaching Methods	Theory + Practicum Lectures/Discussions/Presentations/Case Studies, etc.
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A)1: To sensitize the students about context in which they are embedded i.e. Indian culture and civilization including its Knowledge System and Tradition. CO(A)2: To help student to understand the knowledge, art and creative practices, skills and values in ancient Indian system. CO(A)3: To help to study the enriched scientific Indian heritage. CO(A)4: To introduce the contribution from Ancient Indian system & tradition to modern science & Technology.	
8	Course Outcomes: Student will be able to CO1: Understand and appreciate the rich Indian Knowledge Tradition. CO2: Understand the contribution of Indians in various fields. CO3: Experience increase subject-awareness and self-esteem. CO4: Develop a comprehensive understanding of how all knowledge is ultimately intertwined.	

Syllabus

UNIT I: Introduction

- **Introduction to IKS** (What is knowledge System, Characteristic Features of Indian Knowledge System)
- **Why IKS?** (Macaulay's Education Policy and its impact, Need of revisiting Ancient Indian Traditions)
- **Scope of IKS** (The Universality of IKS (from Micro to Macro), development form Earliest times to 18th Century CE)
- **Tradition of IKS** (Ancient Indian Education System: Home, Gurukul, Pathashala, Universities and ancient educational centres)
- **Relevant sites in the vicinity of the Institute** (Water Management System at Kanheri, Temple Management of Ambernath, etc.)

UNIT II


- Medicine (Ayurveda)
- Alchemy
- Mathematics
- Logic
- Art of Governance (Arthashastra)

UNIT III (Select Any FIVE out of the following)

- Aesthetics
- Town Planning
- Strategic Studies
- Krishi Shastra
- Vyakaran & Lexicography
- Natyashastra
- Ancient Sports
- Astronomy
- Yoga and Wellbeing
- Linguistics
- Chitrasutra
- Architecture
- Taxation
- Banking
- Trade and Commerce

10	<p>Scheme of Examination and Assessment Pattern Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 1:00 hours Format of Question Paper</p> <p>Attempt all questions.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;">Question No</th> <th style="width: 65%;">Nature of Questions</th> <th style="width: 20%;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Q1</td> <td>Attempt any two out of five</td> <td style="text-align: center;">06</td> </tr> <tr> <td style="text-align: center;">Q2</td> <td>Attempt any three out of five</td> <td style="text-align: center;">12</td> </tr> <tr> <td style="text-align: center;">Q3</td> <td>Attempt any three out of fifteen</td> <td style="text-align: center;">12</td> </tr> <tr> <td style="text-align: center;">TOTAL</td> <td></td> <td style="text-align: center;">30</td> </tr> </tbody> </table> <p>Internal Examination: Continuous Evaluation - 20 marks</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;"></th> <th style="width: 65%;">Assessment / evaluation</th> <th style="width: 20%;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1.</td> <td>Class test during lectures (MCQ / Short notes / Match the pairs / Puzzles)</td> <td style="text-align: center;">10</td> </tr> <tr> <td style="text-align: center;">2.</td> <td>Participation in Workshop / Conference / Seminar / Case Study / Field Visit / Certificate Course / Project presentation / Viva</td> <td style="text-align: center;">10</td> </tr> <tr> <td style="text-align: center;">TOTAL</td> <td></td> <td style="text-align: center;">20</td> </tr> </tbody> </table>	Question No	Nature of Questions	Marks	Q1	Attempt any two out of five	06	Q2	Attempt any three out of five	12	Q3	Attempt any three out of fifteen	12	TOTAL		30		Assessment / evaluation	Marks	1.	Class test during lectures (MCQ / Short notes / Match the pairs / Puzzles)	10	2.	Participation in Workshop / Conference / Seminar / Case Study / Field Visit / Certificate Course / Project presentation / Viva	10	TOTAL		20
Question No	Nature of Questions	Marks																										
Q1	Attempt any two out of five	06																										
Q2	Attempt any three out of five	12																										
Q3	Attempt any three out of fifteen	12																										
TOTAL		30																										
	Assessment / evaluation	Marks																										
1.	Class test during lectures (MCQ / Short notes / Match the pairs / Puzzles)	10																										
2.	Participation in Workshop / Conference / Seminar / Case Study / Field Visit / Certificate Course / Project presentation / Viva	10																										
TOTAL		20																										
11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Concise history of science in India- D.M. Bose, S.N Sen, B.V. Subbarayappa. 2. Positive sciences of the Ancient Hindus- Brajendranatha seal, Motilal Banarasidas, Delhi 1958. 3. History of Chemistry in Ancient India & Medieval India, P. Ray- Indian Chemicals Society, Calcutta 1956. 4. Charaka Samhita- a scientific synopsis, P. Ray & H.N Gupta National Institute of Sciences of India, New Delhi 1965. 5. MacDonnell A.A- History of Sanskrit literature. 6. Winternitz M- History of Indian Literature Vol. I, II. 7. Dasgupta S.N & De S.K- History of Sanskrit literature Vol' I. 8. Ramkrishna Mission- cultural heritage of India Vol' I, II. 9. Majumdar R. C & Pushalkar A.D- History & culture of the Indian people, Vol. I, II & III. 10. Keith A.B- History of Sanskrit literature. 																											

Name & Signature of the Dean & Adhoc BoS Chairperson (Interdisciplinary):


 (Dr. Nitin Arekar)



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- I

Title: Cocurricular Course I

**Vertical - 6
Cocurricular Course - 2 Credits**

**with effect from
Academic Year 2025-2026**



Bachelor of Science (Computer Science) (Self-Financing Course)

Semester – II

**Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Vertical – 1

Design and Analysis of Algorithms : 2 Credits

Object Oriented Programming : 2 Credits

Computer Science Practical 2 : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Title: Design and Analysis of Algorithms

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Design and Analysis of Algorithms
Course Code: CHMCSIII

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Design and Analysis of Algorithms course provides a comprehensive study of algorithmic principles, focusing on designing efficient solutions and analyzing their computational performance. It serves as a cornerstone of computer science education, equipping students with essential problem-solving skills to tackle complex computational challenges.</p> <p>Relevance:</p> <p>In today's rapidly advancing field of computer science, the Design and Analysis of Algorithms course holds significant relevance. It empowers students with the analytical and problem-solving skills needed to tackle challenges across various domains, including software engineering and artificial intelligence.</p> <p>Usefulness:</p> <p>This course plays a vital role in developing algorithmic thinking. Students gain the ability to design efficient algorithms, verify their correctness, and assess their performance-key competencies required for building optimized and effective solutions across a wide range of computing applications.</p> <p>Application:</p> <p>The knowledge acquired in this course can be applied across diverse domains — from designing efficient search and sorting algorithms to optimizing resource utilization in areas such as network design and artificial intelligence.</p> <p>Interest:</p> <p>This course captivates students with its intellectual rigor and problem-solving focus, engaging them in analyzing complex problems, designing algorithmic solutions, and evaluating their efficiency, thereby fostering a strong foundation and appreciation for algorithmic thinking.</p> <p>Connection with Other Courses:</p> <p>The Design and Analysis of Algorithms course builds essential links with other areas of computer science, serving as a foundation for advanced studies in data structures, algorithmic complexity, and computational theory, and fostering a comprehensive understanding of computational principles.</p>

		<p>Demand in the Industry:</p> <p>Professionals skilled in algorithm design and analysis are highly sought after across various industries. Sectors such as technology, finance, and healthcare value experts who can craft efficient algorithms to solve complex problems, optimize processes, and improve overall system performance.</p> <p>Job Prospects:</p> <p>Graduates of the Design and Analysis of Algorithms course are well-prepared for diverse career paths, including roles such as software engineer, algorithm developer, data scientist, and research scientist. They are highly valued for their expertise in formulating elegant, efficient, and scalable solutions to complex computational problems.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: Understand the fundamental principles of algorithm design and their role in developing efficient computational solutions.</p> <p>CO(A) 2: Explain the theoretical foundations of basic data structures and their application in algorithmic problem-solving.</p> <p>CO(A) 3: Apply core problem-solving strategies such as searching, sorting, selection, and recursion, and analyze the efficiency of various algorithms.</p> <p>CO(A) 4: Explore and implement major algorithm design paradigms to address diverse computational problems.</p> <p>CO(A) 5: Integrate algorithmic techniques and design strategies to solve real-world challenges effectively and evaluate their performance and scalability.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Understand the importance of algorithm efficiency and evaluate program performance based on the algorithms used.</p> <p>CO2: Recognize and appreciate the role of various data structures in solving computational problems effectively.</p> <p>CO3: Identify problem requirements and select suitable algorithm design principles for optimal solutions.</p> <p>CO4: Apply appropriate algorithms and data structures to address real-world problem scenarios.</p>	

	<p>CO5: Analyze and compare alternative approaches to enhance the efficiency and scalability of software solutions.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Algorithms – Definition and meaning of an algorithm; analysis of algorithms; types of complexity; running time analysis; comparison of algorithms; rate of growth; types of analysis; asymptotic notation including Big-O, Omega (Ω), and Theta (Θ) notations; asymptotic analysis; performance characteristics of algorithms; estimation of running time or number of execution steps on paper; and the idea of computability. • Algorithm Efficiency and Optimization: Introduction to algorithm design efficiency, factors affecting performance, and optimization strategies. • Algorithm Visualization and Tracing: Concept of tracing algorithms using pseudo-code and dry-run analysis to understand flow and complexity. • Recursion – Concept of recursion; comparison between recursion and iteration; applications such as factorial computation, Fibonacci series generation, and the Tower of Hanoi problem; comparative analysis of recursive and iterative approaches. • Basic Sorting Techniques – Concepts and implementation of bubble sort, selection sort, and insertion sort; and comparative analysis of their performance and efficiency. <p>UNIT II:</p> <ul style="list-style-type: none"> • Searching Techniques: Concept of searching algorithms. Linear Search – definition, working, and types. Binary Search – concept, working, and implementation. Comparative analysis of linear and binary search techniques based on efficiency, time complexity, and applications. • Selection Techniques: Introduction to selection techniques. Selection by sorting and partition-based selection algorithms. Finding the <i>k</i>th smallest element in sorted order. Comparative analysis of selection techniques with respect to time complexity, efficiency, and implementation methods. • String Algorithms: Introduction to pattern matching in strings. Study of brute force method for string matching and its stepwise working. Comparative analysis of brute force method with other pattern matching techniques in terms of performance and applicability. • Algorithm Design Techniques: Overview of algorithm classifications, design criteria, and various design techniques. • The Greedy Method: The general method, Knapsack problem, Job sequencing with deadlines, minimum-cost spanning trees – Prim’s and Kruskal’s Algorithms, Optimal storage on tapes, Optimal merge Patterns, Single source shortest paths.

- **Divide and Conquer:** Concept, working principle, advantages, disadvantages, and applications. Implementation using examples like Merge Sort and Strassen's Matrix Multiplication.
- **Dynamic Programming:** Concept and fundamentals of dynamic programming. Advantages and disadvantages of the technique. Applications and implementations using problems such as Fibonacci Series, Factorial of a Number, and Longest Common Subsequence.
- **Backtracking Programming:** Concept of backtracking as a problem-solving approach. Advantages and disadvantages of backtracking. Applications and implementation through the N-Queen Problem.

10	<p>Scheme of Examination and Assessment Pattern Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 01:00 hours</p> <p>Format of Question Paper</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="text-align: center;">Q.No.</th> <th style="text-align: center;">Structure of the Questions</th> <th style="text-align: center;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">3.</td> <td>Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)</td> <td style="text-align: center;">15</td> </tr> <tr> <td style="text-align: center;">4.</td> <td>Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)</td> <td style="text-align: center;">15</td> </tr> <tr> <td colspan="2" style="text-align: center;">Total</td> <td style="text-align: center;">30</td> </tr> </tbody> </table> <p>Internal Examination: Continuous Evaluation – 20 marks</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th style="width: 5%;"></th> <th style="text-align: center;">Assessment / Evaluation</th> <th style="text-align: center;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1.</td> <td>Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes.</td> <td style="text-align: center;">10</td> </tr> <tr> <td style="text-align: center;">2.</td> <td>Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.</td> <td style="text-align: center;">10</td> </tr> <tr> <td colspan="2" style="text-align: right;">Total 20</td> <td></td> </tr> </tbody> </table>	Q.No.	Structure of the Questions	Marks	3.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15	4.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15	Total		30		Assessment / Evaluation	Marks	1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10	2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10	Total 20		
Q.No.	Structure of the Questions	Marks																							
3.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15																							
4.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15																							
Total		30																							
	Assessment / Evaluation	Marks																							
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10																							
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10																							
Total 20																									

11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Data Structure and Algorithm Using Python, Rance D. Ncaise, Wiley India Edition, 2016. 2. Data Structures and Algorithms Made Easy, Narasimha Karumanchi, CareerMonk Publications, 2016. 3. Introduction to Algorithms, Thomas H. Cormen, 3rd Edition, PHI. 4. Introduction to the Design and Analysis of Algorithms, Anany Levitin, Pearson, 3rd Edition, 2011. 5. Design and Analysis of Algorithms, S. Sridhar, Oxford University Press, 2014 6. Introduction to Algorithms: https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/
-----------	---

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Title: Object Oriented Programming

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Object Oriented Programming
Course Code: CHMCSII2

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Object-Oriented Programming Using Java course provides a foundational understanding of object-oriented principles and their practical implementation through the Java programming language. It introduces students to essential software design and development concepts, preparing them to build modular and efficient applications.</p> <p>Relevance:</p> <p>In today's software development landscape, understanding object-oriented programming principles is essential. Java, with its strong OOP foundation and platform independence, is widely used for developing reliable, secure, and scalable applications. This course is therefore highly relevant to the demands of modern software development.</p> <p>Usefulness:</p> <p>This course plays a vital role in developing a solid grasp of core programming paradigms such as encapsulation, inheritance, and polymorphism. It equips students with the ability to design modular, reusable, and maintainable code, fostering the development of efficient and scalable software systems.</p> <p>Application:</p> <p>The concepts covered in this course are directly applied in software development. Students learn to organize programs using classes and objects, enabling the design of efficient, structured, and maintainable applications.</p> <p>Interest:</p> <p>The course engages students through its practical and creative approach. By working on hands-on exercises, learners apply OOP principles to design and implement solutions, fostering strong interest and confidence in software development.</p>

		<p>Connection with Other Courses:</p> <p>This course forms the foundation for various programming and software engineering subjects. It supports advanced learning in software architecture, design patterns, and application development, creating a smooth progression toward more complex programming concepts.</p> <p>Demand in the Industry:</p> <p>Professionals with a strong grasp of object-oriented programming using Java are highly sought after. From software development to embedded systems, industries value individuals who can apply OOP principles to build efficient, modular, and maintainable solutions.</p> <p>Job Prospects:</p> <p>Students completing this course can explore diverse career opportunities, including roles such as software developer, systems analyst, application architect, and embedded systems. These professionals are valued for their ability to design and develop robust, scalable, and maintainable software solutions.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: To enable learners to understand the fundamental concepts and principles of Object-Oriented Programming (OOP).</p> <p>CO(A) 2: To develop the ability to design object-oriented solutions using UML class diagrams and AI-assisted modeling tools.</p> <p>CO(A) 3: To familiarize learners with the syntax and structure of the Java programming language.</p> <p>CO(A) 4: To help learners analyze, implement, and apply various OOP concepts in program development.</p>	

	<p>CO(A) 5: To encourage learners to design and develop programs demonstrating OOP concepts effectively.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: The learner will be able to understand, explain, and demonstrate the fundamental concepts of Object-Oriented Programming (OOP).</p> <p>CO2: The learner will be able to model and interpret UML class diagrams, including key object-oriented relationships, and utilize AI-based tools for automated UML generation from prompts or source code.</p> <p>CO3: The learner will be able to illustrate and apply different types of control statements using Java syntax.</p> <p>CO4: The learner will be able to analyse, implement, and integrate key OOP concepts in program development.</p> <p>CO5: The learner will be able to write and develop programs that effectively demonstrate OOP principles.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Programming Concepts: Overview of the object-oriented programming paradigm, key principles, advantages, and commonly used languages. Comparison of structured and object-oriented approaches, components of an OOP environment (compiler, interpreter, runtime system), with illustration of basic program structure and execution using Java syntax. • Data Types and Operators: Basic data types, Tokens, keywords, variables, rules for naming variables, identifiers, literals, constants, operators, expressions, precedence of operators, type conversion and casting. • Control Structures, Loops, Arrays, and Strings: Decision-making statements – if, if...else, if...else if, switch; loops – while, do...while, for, nested loops; break and continue statements; arrays – types and initialization; and basic string handling and manipulation. • Classes, Objects, Methods, and Structural Concepts: Concept of classes and objects, method definition and invocation, instance and class members, static members, packages, access specifiers, and accessor - mutator methods (encapsulation). • Constructors and Object Lifecycle: Constructors and its types, parameterized constructors, constructor overloading, use of the ‘this’ keyword, and memory management during object lifecycle. <p>UNIT II:</p> <ul style="list-style-type: none"> • Inheritance and Polymorphism: Concept and types of inheritance, use of the super keyword, method overloading and overriding, final members and classes, and dynamic method dispatch.

- **Abstraction and Interfaces:** Concept and need for abstraction, abstract classes and methods, and implementation of abstraction using interfaces.
- **Exception Handling:** Concept and need for exception handling in object-oriented programming, types of exceptions, use of try-catch-finally constructs, throwing exceptions, and creating user-defined exception classes.
- **Inner and Anonymous Classes:** Types of inner classes- static, member, local, and anonymous classes and their uses in simplifying object-oriented design.
- **File Handling and Stream I/O:** Concept of file handling using stream classes, opening and closing files, text and binary file operations, and reading or writing object data in an object-oriented manner.
- **Modeling with UML:** Introduction to UML and class diagrams; modeling relationships including association, aggregation, composition, and generalization–specialization with suitable examples, using AI-based UML tools for automatic generation of UML diagrams from prompts or source code.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q.No.	Structure of the Questions	Marks
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
Total		30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Herbert Schildt, Java The Complete Reference, Eleventh Edition, McGraw-Hill Education, 2020.
2. E. Balagurusamy, Programming with Java- A Primer, Tata McGraw-Hill Education India, 2023.
3. Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press, 2018.
4. Introduction to Programming Using Java by Prof. David J. Eck, Hobart and William Smith Colleges, 8th Edition (2021): <https://math.hws.edu/javanotes/contents-with-subsections.html>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Title: Computer Science Practical 2

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Computer Science Practical 2
Course Code: CHMCSII3

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Computer Science Practical Course in Design and Analysis of Algorithms and Object-Oriented Programming (OOP) using JAVA offers an in-depth understanding of core computational concepts and programming techniques. It combines algorithmic problem-solving with practical implementation through the JAVA language, fostering analytical thinking and coding proficiency.</p> <p>Relevance:</p> <p>In today's rapidly evolving field of computer science, integrating algorithmic design with object-oriented programming is vital. This course enables students to develop essential skills for solving complex problems, designing efficient algorithms, and implementing effective solutions using the OOP approach in JAVA</p> <p>Usefulness:</p> <p>The course plays a crucial role in building a solid foundation in algorithmic thinking and software design. Students learn to evaluate algorithm efficiency, apply OOP principles for modular coding, and develop robust software solutions, thereby strengthening their overall programming expertise.</p> <p>Application:</p> <p>The concepts learned in this practical course have direct application in software development. Students gain hands-on experience in designing and implementing algorithms, analyzing their performance, and developing applications using object-oriented principles in JAVA.</p> <p>Interest:</p> <p>The course engages students through its practical and project-based approach. By applying algorithmic strategies, designing class hierarchies, and implementing</p>

		<p>solutions in JAVA, learners develop a strong interest in problem-solving and software development.</p> <p>Connection with Other Courses:</p> <p>This practical course is closely aligned with other core computer science subjects. It builds a solid foundation for advanced learning in areas such as algorithmic complexity, data structures, software engineering, and object-oriented programming, delivering a balanced and comprehensive education</p> <p>Demand in the Industry:</p> <p>Professionals skilled in algorithmic design and object-oriented programming using JAVA are highly sought after across various industries. Sectors such as software development, technology, and finance value these competencies for developing efficient, reliable, and scalable software systems.</p> <p>Job Prospects:</p> <p>Graduates of this practical course have a wide range of career opportunities, including roles such as software engineer, algorithm developer, systems analyst, and application developer. They are highly valued for their ability to design and implement algorithmically efficient, modular, and maintainable software solutions.</p>
2	Vertical 1	Major
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1: Analyse and implement algorithms to solve common computational problems efficiently.</p> <p>CO(A) 2: Apply algorithmic strategies such as divide and conquer, dynamic programming, and greedy methods for optimization.</p> <p>CO(A) 3: Understand and apply object-oriented programming principles including encapsulation, inheritance, and polymorphism.</p> <p>CO(A) 4: Design and implement classes and objects to develop modular and reusable software components.</p>	

	<p>CO(A) 5: Utilize access specifiers and other OOP features to ensure secure and maintainable program design.</p>
8	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Design, implement, and evaluate algorithms for various computational problems using appropriate strategies.</p> <p>CO2: Analyse and compare the time and space complexities of algorithms for efficiency optimization.</p> <p>CO3: Apply divide and conquer, dynamic programming, and greedy techniques to solve complex problems.</p> <p>CO4: Design and implement classes, objects, and methods to build modular Java programs.</p> <p>CO5: Apply advanced OOP concepts such as inheritance, polymorphism, and abstraction, encapsulation to enhance code reusability and scalability.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1: Linear and Binary Search</p> <p>a) Implementation of linear and binary search algorithms on a list and comparison of their performance.</p> <p>Practical 2: Sorting Algorithms</p> <p>a) Implementation of basic sorting algorithms such as Bubble Sort, Selection Sort, and Insertion Sort, and analysis of their efficiency.</p> <p>Practical 3: Nth Maximum / Minimum Element</p> <p>a) Implementation of an algorithm to determine the Nth maximum or minimum element in a list.</p> <p>Practical 4: Recursion</p> <p>a) Implementation of recursive algorithms for problems such as Factorial, Fibonacci Series, and Tower of Hanoi.</p> <p>Practical 5: String Pattern Matching</p> <p>a) Implementation of string pattern matching algorithms using the brute force method or other approaches</p>

Practical 6: Greedy Algorithm

- a) Application of the Greedy Technique to solve problems such as File Merging and Coin Change

Practical 7: Divide and Conquer Technique

- a) Implementation of Divide and Conquer algorithms including Merge Sort and Strassen's Matrix Multiplication.

Practical 8: Dynamic Programming

- a) Implementation of algorithms using dynamic programming for solving problems like Fibonacci Series and Longest Common Subsequence.

UNIT II:

Practical 1: Basic concepts and Introduction to Classes:

- a) Demonstrating data types and type conversions.
- b) Exploring operator precedence and associativity.
- c) Creating classes, objects, and invoking methods to observe object behavior.

Practical 2: Conditional statements and Looping constructs:

- a) Writing Programs using conditional statements (if, if...else, switch) for decision-making.
- b) Implementing programs using looping constructs (for, while, do...while) for repetitive operations.
- c) Demonstrate the use of break and continue statements within loop execution.

Practical 3: Arrays and Strings

- a) Implementing programs to declare, initialize, and display elements of a one and multi-dimensional array.
- b) Demonstrating immutability of strings.
- c) Exploring String class methods and demonstrating string comparison.

Practical 4: Constructors, Packages, Access Specifiers and Encapsulation

- a) Using constructors to initialize objects.

- b) Applying the ‘this’ keyword and static members.
- c) Implementing programs using predefined packages.
- d) Creating and importing user-defined packages across multiple classes.
- e) Demonstrating the use of access specifiers to control data visibility and accessibility.
- f) Demonstrating encapsulation using getters and setters.

Practical 5: Exception Handling:

- a) Implementing programs to handle predefined exceptions using try, catch, and finally blocks.
- b) Demonstrating exception handling for throwing exceptions.
- c) Creating and handling user-defined (custom) exceptions.

Practical 6: Inheritance and Polymorphism:

- a) Reusing code through inheritance and the ‘super’ keyword.
- b) Implementing constructor chaining using base and derived classes.
- c) Programs to illustrate method overloading.
- d) Programs to illustrate method overriding.

Practical 7: Abstraction and Interfaces:

- a) Developing programs on Abstract classes
- b) Developing programs on interfaces.

Practical 8: File Handling:

- a) Illustrating the concept of stream classes and developing programs to perform file read/write operations.

Supplementary Exercise: Demonstration of GitHub Copilot AI tool for programming assistance, including code suggestion, auto-completion, and debugging support in VS Code.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly

certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Algorithms, 4th Edition by Robert Sedgewick and Kevin Wayne, Addison-Wesley, 2011: <https://introcs.cs.princeton.edu/java/40algorithms/>
2. Problem Solving with Algorithms and Data Structures using Python by Brad Miller and David Ranum
<https://runestone.academy/ns/books/published/pythonds/index.html>
3. Herbert Schildt, Java The Complete Reference, Eleventh Edition, McGraw-Hill Education, 2020.
4. Java: A Complete Practical Solution by Swati Saxena — BPB Publications, 2019
5. Java SE API Documentation by Oracle <https://docs.oracle.com/en/java/javase/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

**Vertical - 2
Minor Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- II

**Title: Applied Logic, Analytical Reasoning and
Coding Techniques**

**Vertical – 2
Minor – 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Applied Logic, Analytical Reasoning and Coding Techniques
Course Code: CHMMATHIII3

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>This course introduces students to the foundational and applied aspects of logical reasoning and data interpretation, focusing on structured thinking, pattern recognition, spatial reasoning, relational analysis, coding–decoding, and data sufficiency. These competencies form the backbone of problem-solving across disciplines, including mathematics, computer science, and data science.</p> <p>Emphasis is placed on developing the ability to analyze numerical, symbolic, and verbal patterns; interpret and represent relational and directional information; and evaluate the adequacy of data in real-world and mathematical contexts. The course also strengthens skills essential for competitive and entrance examinations such as Banking, Management Aptitude Tests, UPSC/MPSC, SET/NET, GRE/GMAT, among others, where logical thinking and data interpretation constitute core testing components.</p> <p>Through graded difficulty levels and practice-oriented learning, students will gain the analytical capacity, reasoning accuracy, and decision-making clarity required to tackle diverse logical and data-driven problems.</p>
2	Vertical 2	Minor
3	Type & Teaching Method	Theory + Practicum (Lectures / Problem Solving / Discussion / Presentation / Case Study / Demonstration etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<p>This course reinforces essential mathematical concepts acquired during school education while engaging learners with advanced, indirect, and analytically challenging problems. It is designed to cultivate critical thinking, strengthen logical and reasoning abilities, and encourage deeper</p>

	<p>intellectual engagement with numerical, symbolic, and relational information. By developing these core cognitive skills, the course prepares learners to approach complex problem-solving tasks with confidence and enhances their competence for various competitive and entrance examinations.</p> <p>CO(A) 1: To strengthen foundational abilities in pattern recognition, symbolic manipulation, and sequential reasoning through structured exercises in numerical, alphabetic, and symbolic series.</p> <p>CO(A) 2: To develop analytical and deductive reasoning skills by engaging with classification tasks, relational mapping, directional problems, and spatial reasoning scenarios.</p> <p>CO(A) 3: To build proficiency in encoding and decoding information using alphabetic, numeric, symbolic, and substitution-based systems relevant to mathematical reasoning and computational logic.</p> <p>CO(A) 4: To cultivate the ability to assess data adequacy and interpret information presented in varied forms, enabling learners to determine the minimum required data for decision-making in mathematical and real-life contexts.</p> <p>CO(A) 5: To prepare learners for advanced logical reasoning tasks encountered in academic, professional, and competitive examination settings by nurturing critical thinking, precision, and systematic problem-solving abilities.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of the course, learners will be able to:</p> <p>CO1: Analyze numerical, alphabetical, and symbolic sequences to identify underlying patterns and determine missing or predicted elements.</p> <p>CO2: Apply classification and logical comparison skills to identify inconsistencies or odd elements within a given set of numbers, symbols, or objects.</p> <p>CO3: Construct and interpret relational and directional representations—including family trees, generational diagrams, and spatial pathways—to solve reasoning problems.</p> <p>CO4: Apply coding–decoding principles using alphabetic, numeric, symbolic, and substitution-based methods to encode information or interpret coded messages.</p> <p>CO5: Evaluate the sufficiency of given information in mathematical or real-life scenarios and determine the minimum additional data required to reach a valid conclusion.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I: Foundations of Logical Thinking</p> <p>1. Number, Letter, and Symbol Series</p> <ul style="list-style-type: none"> • Identification of patterns in numerical sequences, including differences, ratios, powers (squares and cubes), and recursive relations. • Completion of missing terms at the beginning, middle, or end of a given sequence. • Analysis of sequences comprising letters or symbols; recognition of embedded patterns and prediction of missing elements.

2. Analytical Reasoning: Odd-One-Out

- Determining the element that does not conform to the underlying rule governing a sequence of numbers.
- Identifying inconsistencies in sequences composed of letters or symbols by observing structural or logical deviations.

3. Relational Reasoning

- Understanding common family and social relationship terminologies: parent, child, grandparent, sibling, spouse, nephew, niece, cousin, etc.
- Construction of family diagrams or relational trees using standardized notation:
 - a) **Square** for male, **circle** for female, **triangle** for unspecified/undetermined gender
 - b) **Double arrow** (\leftrightarrow) for siblings
 - c) **Equal sign** ($=$) for marital/spousal relationships
- Interpretation of information across generations through vertical alignment of relational structures.

UNIT II: Applied Logical Thinking and Data Interpretation

1. Direction Sense and Spatial Reasoning

- Understanding and identifying the eight cardinal and intercardinal directions.
- Interpreting angular relationships between directions.
- Application of standard Pythagorean triplets (3-4-5, 6-8-10, 5-12-13, etc.) and 45° – 45° – 90° triangles in solving distance and movement problems involving perpendicular paths.

2. Coding and Decoding Techniques

- Introduction to various coding mechanisms:
 - a) Alphabet-based coding
 - b) Numerical coding
 - c) Symbol coding
 - d) Value-based and substitution coding
- Deciphering coded messages and constructing logical mappings between original and encoded information.

3. Data Sufficiency

- Understanding the concept and principles of data sufficiency, focusing on how to determine whether the information provided is adequate to reach a definite conclusion.
- Identifying whether a problem is solvable, unsolvable, or conditionally solvable based on the given data, and recognizing situations where additional information is required.
- Working with problems drawn from elementary mathematics and everyday

contexts, such as:

- Determining whether the **area or perimeter** of a geometric figure (triangle, quadrilateral, circle) can be computed with the given measurements.
- Checking whether the **value of an unknown variable** in simple algebraic equations can be determined with the available conditions.
- Assessing whether the **age of a person** or the **relation between two individuals** can be concluded from partial relational information.
- Evaluating whether the **time, distance, or speed** in a basic motion scenario can be derived from the data provided.
- Judging whether a **quantity comparison** (e.g., which number is larger, which container holds more water) can be made without exact values.
- Emphasis is placed on identifying the minimum additional information required for a conclusive solution, rather than computing the final numerical answer.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

Q. No.	Structure of the Questions	Marks
1	Attempt ANY THREE out of SIX. (Each question of 5 marks, based on unit 1)	15
2	Attempt ANY THREE out of SIX. (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Class Test- It should be conducted using any Learning Management System such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes.	10
2.	Project, Self-Learning Evaluation, Presentation, etc.	10
	Total	20

11

REFERENCES:

10. **Aggarwal, R. S.** *A Modern Approach to Verbal & Non-Verbal Reasoning.* S. Chand Publishing.
11. **Gupta, A. K.** *Logical and Analytical Reasoning.* Ramesh Publishing House.
12. **Aggarwal, R. S.** *Quantitative Aptitude for Competitive Examinations.* S. Chand

Publishing.

13. **Sinha, Nishit K.** *Reasoning for Competitive Examinations.* Pearson Education.

14. **Jaikishan & Premkishan.** *How to Crack Test of Reasoning in All Competitive Exams.* Arihant Publications.

15. **Gourav Gupta.** *Logical Reasoning & Data Interpretation.* McGraw Hill / Other Competitive Exam Publishers.

16. *The Pearson Guide to Logical Reasoning & Data Interpretation.* Pearson Education.

Online references:

1. <https://cracku.in/cat-previous-papers>

2. <https://www.selfstudys.com/>

3. <https://testbook.com/reasoning/coding-decoding>

4. <https://testbook.com/reasoning/data-sufficiency-reasoning>

5. <https://www.indiabix.com/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

**Vertical - 3
Open Elective Course: 2+2 Credits**

**with effect from
Academic Year 2025-2026**



HSNC Board's

Smt. Chandibai Himathmal Mansukhani College

(Autonomous)

(Affiliated to the University of Mumbai)

University College Code: 217 | JD Office: T14



Faculty of Interdisciplinary

Vertical 3: List of Open Elective Skill Based Courses for First Year: Semester – II

Sr. No.	Nomenclature of the Paper
1	Cyber and Digital Safety
2	Audio -Video Editing Foundation for Graphics Design and Basics of Animation II
3	Basic Tools of AI for Economics and Education - II
4	English for Professional and Corporate World
5	Urbanisation and Real Estate
6	Business of Travel and Tours-II
7	Managing Family Wealth through Family Office
8	Interactive Web Design using Java Script
9	Basics of Nutrition II
10	Reels Production and Creator Branding
11	Performing Art
12	Data Analysis with Advanced Excel
13	Advanced Political Communication and Media Skills
14	Stress Management
15	Social Media and Society Identity, Power and Digital Citizenship
16	Mushroom Cultivation Training and Trading Level
17	Yoga and Fitness II
18	Basic Perfumery Course (Level-II)
19	Soft Skills II-Personal and Interpersonal Effectiveness
20	Beautician - Strategic Business Planning-II
21	Current Trends of Fashion Designing- Financial Perspective Level-II
22	Basic Accounting-II
23	Digital Marketing II
24	Online Trading and Stock Market-II
25	Event Management Course in Sindhi



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Vertical – 4

Vocational Skill Course (VSC) : 2 Credits

Skill Enhancement Course (SEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Title: Web Technologies

**Vertical - 4
Vocational Skill Course (VSC) - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Web Technologies
Course Code: CHMCSII6

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Web Technologies course offers an in-depth exploration of the core technologies that power the visual and interactive aspects of the web. Covering HTML, CSS, JavaScript, XML, and PHP, the course equips learners with the technical and creative skills needed to develop dynamic, user-friendly, and visually engaging websites.</p> <p>Relevance:</p> <p>In today's digital era, web technologies play a crucial role in shaping the online experience. This course holds strong relevance as it introduces learners to the core languages and frameworks that form the foundation of modern web development.</p> <p>Usefulness:</p> <p>This course is highly valuable for individuals seeking to build responsive, user-friendly, and visually engaging websites. It provides hands-on experience in structuring web content, designing layouts, and integrating interactive elements to enhance user experience.</p> <p>Application:</p> <p>The knowledge gained in this course is directly applicable to real-world web development projects. Learners design and develop complete websites, utilizing HTML for structure, CSS for styling, JavaScript for interactivity, XML for data handling, and PHP for server-side scripting.</p> <p>Interest:</p> <p>The creative and practical nature of web technologies makes the learning experience highly engaging. Through hands-on projects, learners actively design and develop websites, cultivating a strong interest in crafting visually appealing and interactive online experiences.</p>

		<p>Connection with Other Courses:</p> <p>This course builds a strong foundation that connects seamlessly with other areas of web development and computer science. It prepares learners for advanced studies in full-stack development, database management, and server-side programming, strengthening their overall technical proficiency.</p> <p>Demand in the Industry:</p> <p>Professionals with expertise in web technologies are highly sought after across industries such as e-commerce, technology, and media. Organizations actively look for individuals capable of developing user-friendly, visually appealing, and interactive websites that enhance online presence and drive user engagement.</p> <p>Job Prospects:</p> <p>Graduates of the Web Technologies course have diverse career opportunities in the digital domain. Potential roles include web designer, front-end developer, UI/UX designer, and web content manager. These professionals are valued for their expertise in designing visually appealing, functional, and user-centered web interfaces.</p>
2	Vertical 4	VSC (Vocational Skill Course)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Understand the fundamental concepts and components of web technologies.</p> <p>CO(A) 2. Explore the structure and styling of web pages using HTML and Cascading Style Sheets (CSS).</p> <p>CO(A) 3. Develop dynamic and interactive websites using JavaScript and jQuery.</p> <p>CO(A) 4. Implement data exchange and manipulation among web applications using XML.</p> <p>CO(A) 5. Apply PHP for server-side programming to build functional and data-driven web applications.</p>	

<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Design valid, well-structured, scalable, and meaningful web pages using modern and emerging web technologies.</p> <p>CO2: Understand and adapt web designs for different platforms, devices, display resolutions, viewports, and browsers.</p> <p>CO3: Develop and implement client-side scripting using JavaScript and jQuery, along with server-side programming for dynamic web applications.</p> <p>CO4: Design and develop database-driven websites.</p> <p>CO5: Utilize XML to create markup languages for data-centric and document-centric web applications.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT-I</p> <p>Practical 1: Basic HTML Tags – I: Learn basic HTML tags for structuring a webpage</p> <ol style="list-style-type: none"> a) Document Structure Tags b) Basic Text Formatting Tags c) List Tags d) Image and Image Maps Tags <p>Practical 2: Basic HTML Tags – II: Explore advanced HTML tags for navigation, data presentation, and user input.</p> <ol style="list-style-type: none"> a) Navigation Tags b) Table Tags c) Form and Form Elements Tags d) Embedded Multimedia Elements Tags <p>Practical 3:</p> <p>I) Traditional CSS</p> <ol style="list-style-type: none"> a) Inline, Internal, and External CSS b) CSS Selectors (.class, #id, element) c) The Box Model (margin, border, padding, content) d) Key Properties: <ul style="list-style-type: none"> • Color & Background: color, background-color, background-image • Text & Font Styling: font-family, font-size, text-align • Box Layout: margin, padding, border • Simple Layout with Flexbox: display: flex <p>II) Tailwind CSS:</p> <ol style="list-style-type: none"> a) Introduction to Tailwind b) Basic Styling with Tailwind: Text, Background & Borders, Spacing, Hover States c) Responsive Design: Breakpoints, Layout with flex and grid classes d) Typography & Buttons: Styling forms and buttons

	<p>Practical 4: JavaScript – I: Learn basic JavaScript concepts for dynamic behavior and user interaction.</p> <ol style="list-style-type: none"> Document Object Model (DOM) Performing Mathematical Operations JavaScript Timing and Pop-up Boxes <p>Practical 5: JavaScript – II: Explore advanced JavaScript features for validation and client-side data management.</p> <ol style="list-style-type: none"> String, RegExp, Math, Date Objects Validating various Form Elements Store and Retrieve Cookies <p>UNIT-II</p> <p>Practical 6: jQuery: Faster web development and dynamic interaction</p> <ol style="list-style-type: none"> Introduction to jQuery DOM Manipulation with jQuery Event Handling with jQuery Animations and Effects with jQuery Ajax with jQuery (for dynamic page updates) <p>Practical 7: XML: Understand XML structure and how to apply CSS and XSL for presentation.</p> <ol style="list-style-type: none"> Internal DTD using CSS External DTD using CSS XML using XSL External DTD using XSL <p>Practical 8: PHP – I: Learn the basics of PHP scripting for dynamic and interactive web pages.</p> <ol style="list-style-type: none"> Retrieving data from HTML forms Performing certain mathematical operations Working with Arrays <p>Practical 9: PHP – II: Understand file handling and database operations in PHP for server-side scripting.</p> <ol style="list-style-type: none"> Working with Files and Directories Working with Databases 												
10	<p style="text-align: center;">Scheme of Examination and Assessment Pattern Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 02:00 hours Format of Question Paper</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Question</th> <th>Practical Questions Based on</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">Q.1.</td> <td>Unit – I</td> <td style="text-align: center;">15</td> </tr> <tr> <td style="text-align: center;">Q.2.</td> <td>Unit – II</td> <td style="text-align: center;">15</td> </tr> <tr> <td colspan="2"></td> <td style="text-align: right;">Total 30</td> </tr> </tbody> </table>	Question	Practical Questions Based on	Marks	Q.1.	Unit – I	15	Q.2.	Unit – II	15			Total 30
Question	Practical Questions Based on	Marks											
Q.1.	Unit – I	15											
Q.2.	Unit – II	15											
		Total 30											

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11**REFERENCES:**

1. HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed, Dreamtech Press, 2016
2. Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India, 2018
3. PHP: A Beginners Guide, Vikram Vaswani, TMH
4. HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY, 2011
5. Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'Reilly, 2018
6. PHP, MySQL, JavaScript & HTML5 All-in-one for Dummies, Steve Suehring, Janet Valade Wiley, 2018
7. <https://developer.mozilla.org/en-US/docs/Web>
8. <https://v2.tailwindcss.com/docs/installation>
9. <https://api.jquery.com>
10. <https://www.php.net/manual/en/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

**Title: Database Management Systems using
PL/SQL**

**Vertical - 4
Skill Enhancement Course (SEC) - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Database Management Systems using PL/SQL

Course Code: CHMCSI7

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Database Management Systems (DBMS) using PL/SQL course offers an in-depth study of the concepts and techniques involved in effective database management. It equips learners with the knowledge and practical skills to design, develop, and maintain reliable and efficient database systems using the capabilities of the PL/SQL language.</p> <p>Relevance:</p> <p>In today's information-driven world, databases form the foundation of modern applications. This course holds great relevance as it focuses on PL/SQL, a powerful procedural language that enables seamless interaction with Oracle databases, one of the most widely used and robust database management systems in the industry.</p> <p>Usefulness:</p> <p>This course is essential for individuals aspiring to develop strong proficiency in database management. It trains learners to leverage the power of PL/SQL for efficient data storage, retrieval, and manipulation, thereby enhancing the overall functionality, efficiency, and performance of database systems.</p> <p>Application:</p> <p>The knowledge and skills gained in this course have direct applications in real-world environments. Learners design and implement efficient database structures, develop PL/SQL scripts for data manipulation, and optimize database performance to ensure the smooth and effective functioning of data-driven applications.</p> <p>Interest:</p>

		<p>The experiential and problem-solving approach of working with databases and PL/SQL makes the learning process highly engaging. Through practical exercises, learners design, develop, and manage databases, cultivating a strong interest in achieving efficient data storage, retrieval, and management.</p> <p>Connection with Other Courses:</p> <p>This course builds strong linkages with related areas such as database management, data analytics, and software development. It lays a solid foundation for advanced studies in database optimization, data warehousing, and application development, enabling deeper expertise in data-driven technologies.</p> <p>Demand in the Industry:</p> <p>Professionals skilled in database management using PL/SQL are highly sought after across industries such as finance, healthcare, and e-commerce. Organizations value experts who can design, implement, and maintain databases that ensure data integrity, security, and optimal performance.</p> <p>Job Prospects:</p> <p>Graduates of the DBMS using PL/SQL course have diverse career opportunities across various industries. Potential roles include database administrator, SQL developer, data analyst, and database architect. These professionals are highly valued for their expertise in designing, developing, and managing databases that are essential to organizational efficiency and success.</p>
2	Vertical 4	SEC
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To develop an understanding of the fundamental concepts and techniques used in data management.</p> <p>CO(A) 2. To gain insight into the organization, storage, and retrieval of data for effective management.</p>	

	<p>CO(A) 3.To learn about widely used systems and tools applied for data implementation and utilization.</p> <p>CO(A) 4.To understand the principles of maintaining data consistency and reliability in various system environments.</p> <p>CO(A) 5.To develop a clear understanding of transaction management and crash recovery for ensuring data integrity</p>
8	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Master the concepts and applications of stored procedures, functions, cursors, and triggers in database programming.</p> <p>CO2: Develop skills to manage and manipulate data effectively using PL/SQL.</p> <p>CO3: Utilize collections and records in PL/SQL efficiently.</p> <p>CO4: Understand the principles and implementation of transaction management to ensure data consistency and integrity.</p> <p>CO5: Learn and apply techniques for crash recovery to maintain reliability and robustness in database systems.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1: Sequential and Looping Statements: Fundamental structure programs using sequential execution and simple loops.</p> <ol style="list-style-type: none"> a) Sequential Statements b) Unconstrained Loops <p>Practical 2: Static and Dynamic SQL: Learn the difference between static and dynamic SQL and how to execute both.</p> <ol style="list-style-type: none"> a) Executing Static SQL Queries within PL/SQL Blocks b) Using Dynamic SQL with EXECUTE IMMEDIATE <p>Practical 3: Conditional Control Statements: Understand and implement conditional logic.</p> <ol style="list-style-type: none"> a) IF-THEN Statements b) IF-THEN-ELSE Statements c) IF-THEN-ELSIF-THEN Statements d) CASE Statements <p>Practical 4: Iterative Control Statements: Learn to control program flow using iteration constructs.</p> <ol style="list-style-type: none"> a) WHILE-LOOP Statements b) FOR-LOOP Statements <p>Practical 5: GOTO and NULL Statements: Explore additional control mechanisms such as GOTO and NULL statements.</p> <ol style="list-style-type: none"> a) GOTO Statements in PL/SQL b) NULL Statements in PL/SQL

Practical 6: Procedures: Learn to create, declare, and execute procedures to modularize code.

- a) Create or Replace a procedure and call procedure
- b) Define procedure to insert data
- c) A forward declaration of procedure

UNIT-II:

Practical 7: Sequences: Understand the concept of sequences and their use in generating unique numbers.

- a) Creating and Managing Sequences
- b) Using Sequences in PL/SQL Blocks

Practical 8: Functions: Learn how to define and utilize functions for modular and reusable code.

- a) Define a function
- b) Call function in DBMS_OUTPUT.PUT_LINE
- c) Define and use function in SELECT clause
- d) Call function and store the return value to a variable

Practical 9: Recursive Functions: Understand recursion and implement recursive functions.

Practical 10: Triggers: Learn to create and manage triggers.

- a) Row-Level Triggers
- b) BEFORE and AFTER Triggers

Practical 11: Transactions: Understand transaction control for ensuring data integrity and consistency.

- a) COMMIT, ROLLBACK, and SAVEPOINT Commands
- b) Handling Errors Using Transaction Control Statements

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Mastering PL/SQL Through Illustrations: From Learning Fundamentals to Developing Efficient PL/SQL Blocks, Dr. B. Chandra, BPB Publication, 2020
2. Oracle PL/SQL Training Guide., Training guide, BPB Publications, 2016
3. Raghu Ramakrishnam, Gehrke, Database Management Systems, McGraw Hill, 3rd Edition, 2014
4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6th Edition 2019
5. Ivan Bayross, SQL, PL/SQL -The Programming language of Oracle, B.P.B. Publications 2009
6. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems, Pearson Education, 2008
7. https://docs.oracle.com/cd/B13789_01/appdev.101/b10807.pdf
8. https://www.oracle.com/technetwork/database/features/plsql/overview/sample2174_ch2-129586.pdf

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**First Year B. Sc.
(Computer Science)**

Semester- II

Vertical – 5

Ability Enhancement Course (AEC) : 2 Credits

Value Education Course (VEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year B.Sc.

Semester- II

Vertical – 5

Ability Enhancement Course (English)

2 Credits

**with effect from
Academic Year 2025-2026**

Title: Advanced English for Workplace and Academic Communication
Course Code: CHMBSCAECII

Sr. No.	Heading	Particulars
1	Description of the Course:	In an increasingly competitive academic and professional landscape, learners require advanced communication skills that enable clarity, precision, critical thinking, and professionalism. This course focuses on practical, real-world communication abilities needed for college-level academic work, job applications, workplace collaboration, and digital interactions. Through hands-on tasks, real-world assignments, and communication practice, learners become adept in using English confidently and appropriately in diverse settings.
2	Vertical 5	AEC: Advanced English for Workplace and Academic Communication
3	Type Teaching Methods:	Theory+ Practicum (Lecture/ Discussion/ Presentation/ Reading sessions/ Worksheets/ etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<p>CO(A)1: To develop advanced communication skills required for academic and professional success.</p> <p>CO(A)2: To train learners in report writing, summary writing, and formal documentation.</p> <p>CO(A)3: To enhance proficiency in digital and virtual communication platforms.</p> <p>CO(A)4: To strengthen presentation, interview, and workplace communication skills.</p> <p>CO(A)5: To build confidence in expressing ideas clearly to varied audiences.</p>
8	Course Outcomes:	<p>After completing this course, learners will be able to:</p> <p>CO-1: Demonstrate clarity, precision, and professionalism in communication.</p> <p>CO-2: Interpret and summarize written texts, visuals, and data accurately.</p> <p>CO-3: Prepare well-structured reports, emails, and professional documents.</p>

CO-4: Use digital tools and virtual communication etiquette effectively.

CO-5: Communicate confidently in interviews, presentations, and teamwork situations

9

Syllabus

UNIT I: Communication for Academic & Professional Settings (15 Hours)

A. Communication in Academic & Workplace Contexts

1. Features of formal communication
2. Audience-centered communication
3. Ethics in communication: integrity, attribution, clarity
4. Explaining concepts in simple and clear language
5. Interpreting graphs, charts, tables, and infographics
6. Summarizing data concisely

B. Grammar & Style for Professional Writing

1. Tone: formal, neutral, objective
2. Avoiding redundancy and ambiguity
3. Active vs. passive structures
4. Editing, revising, and proofreading techniques

UNIT II: Practical Documentation & Employability Skills (15 Hours)

1. Report writing (academic/field-based/observational)
2. Project summary reports
3. Preparing short presentations
4. Creating informational posters or digital slides
5. Writing a formal complaint or request email
6. Creating a short informational or awareness write-up

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

All questions are compulsory:

Question No	Nature of Questions	Marks
Q. 1	Short Notes (Attempt any 3 out of 5) - Unit 1 OR Attempt Essay Type question. (1 out of 2) - Unit 1	15
Q. 2	Short Notes (Attempt any 3 out of 5) - Unit 2 OR	15

	Attempt Essay Type question. (1 out of 2) - Unit 2	
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

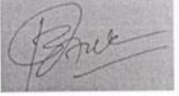


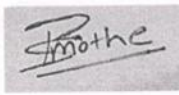
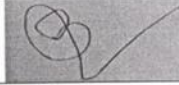
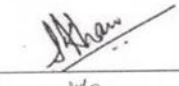
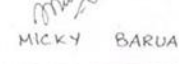

	Assessment / evaluation	Marks
1.	Assignments on any one of the following topics: Academic/Scientific Report/ Informational Poster / Digital Infographic (Students are required to use AI assistance in the preparation of their drafts. Eg: Notion AI, Otter.ai, Grammarly, Google Gemini, Canva, Piktochart, etc)	15
2.	Class Attendance and Participation	05
	Total	20

11

References:

1. Adler, Ronald B., et al. *Understanding Human Communication*. 15th ed., Oxford UP, 2021.
2. Bailey, Stephen. *Academic Writing: A Handbook for International Students*. 5th ed., Routledge, 2018.
3. Cargill, Margaret, and Patrick O'Connor. *Writing Scientific Research Articles*. Wiley-Blackwell, 2013.
4. Eastwood, John. *Oxford Guide to English Grammar*. Oxford UP, 2005.
5. Gerson, Sharon J., and Steven M. Gerson. *Technical Communication: Process and Product*. 9th ed., Pearson, 2021.
6. Hewings, Martin. *Advanced Grammar in Use*. Cambridge UP, 2013.
7. Kumar, Sanjay, and Pushp Lata. *Communication Skills*. 2nd ed., Oxford UP, 2018.
8. McCarthy, Michael, and Felicity O'Dell. *Academic Vocabulary in Use*. Cambridge UP, 2008.
9. Nordquist, Richard. *The Essentials of English Grammar*. McGraw-Hill Education, 2016.
10. Seely, John. *The Oxford Guide to Writing and Speaking*. Oxford UP, 2005.
11. Zinsser, William. *On Writing Well: The Classic Guide to Writing Nonfiction*. Harper Perennial, 2016.

Syllabus Committee:

Sr. No	Name of the Faculty	Designation and College	Signature
1.	Prof. (Dr.) Kailas Aute	Professor & Head, Dept. of English, Smt. CHM College	
2.	Prof. (Dr.) B. R. Hiramani,	(VC Nominee, University of Mumbai) Pancham Khemraj College, Sawantwadi	
3.	Prof. (Dr.) Vikas Raskar	(Subject Expert outside University) Hutatma Rajguru Mahavidyalay, Rajguru Nagar, Khed, (Affiliated to Savitribai Phule University)	
4.	Prof. (Dr.) Prashant Mothe	(Subject Expert outside University) Aadarsh Mahavidyalay, Umerga, Dharashiv, (Affiliated to Dr. Baba Saheb Ambedkar Marathwada University)	
5.	Mr. Ananda Pandhare	Asst. Professor, Dept. of English, Smt. CHM College	
6.	Ms. Sana Khan	Asst. Professor, Dept. of English, Smt. CHM College	
7.	Dr. Micky Barua	Faculty Vidyalankar Institute of technology, Alumni Member	 MICKY BARUA
8.	Ms. Sofy Verghese	Accenture, Industry Representative	

Name & Signature of the Ad-hoc BoS Chairperson: Prof. (Dr.) Kailas Aute



Name & Signature of the Dean: Prof. (Dr.) Nitin Arekar



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- II

Title: Communication Skills in Sindhi

Vertical – 5

Ability Enhancement Course – 2 Credits

**with effect from
Academic Year 2025-2026**

Title: Communication Skills in Sindhi
COURSE CODE: CHMSINIAEC

Sr. No.	Heading	Particulars
1	Description the Course:	Communication is the core component of commerce and trade. In communication, language plays very significant role. If a student has mastered the skills of language, undoubtedly, he or she would be able to communicate in the best manner. In this course basic part of Sindhi language would be taught based on the NEP 2020. Innovative approaches like critical thinking, creative mind, and use of technology will lead to communicating and participating with different groups. The vocabulary section would be given prominence. The course would be in the Devanagari script so that it can attract majority of the students. Even non-Sindhi students shall have opportunity to adopt this course.
2	Vertical 1	AEC – Ability Enhancement Course
3	Type	Theory + Practicum (Teaching Method: Lecture/ Discussion/Reading)
4	Credit	2 credits (1 credit = 15 hours for theory or 30 hours of Practical work in a semester)
5	Hours allotted	30 Hours
6	Marks allotted	50Marks
7	Course Objectives: After successful completion of this course: CO(A) 1: The learner will get understanding of communication skills. CO(A) 2: The learner will understand how to accurate the pronunciation of special words in Sindhi CO(A) 3: The learner will improve the conversation skill in Sindhi. CO(A) 4: The learner will become best communicator in Sindhi language	
8	Course Outcomes: Student will be able to CO1: Know the basic special features of Sindhi language. CO2: Understand communication skills. CO3: Knowing the conversation with businessmen and customers CO4: Know the etiquettes with parents, relatives, friends and others in effective way	

Communication Skills in Sindhi**UNIT I: Everyday & Professional Communication in Sindhi**

- Daily Life Conversation Skills: Greetings and introductions, asking for information, making requests/giving instructions, small talk in simple Sindhi
- Workplace & Business Communication: Customer interaction, Office enquiries, permissions, complaints, Bank/shop/travel conversation, Basic telephone conversation, Short Event Reports

UNIT II: Digital & Modern Communication in Sindhi

- Email & Message Writing: Professional emails, WhatsApp/SMS etiquette, Announcements, reminders
- Product and Promotive Communication: Short ads, Taglines & slogans, Simple product description, Pitch Presentation, Interview / Communication for Market

Syllabus

Communication Skills in Sindhi

यूनिट १: रोज़मरह जी जिंदगी में गुफ्तगू (सिंधीअ में):

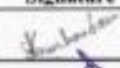

- वाकुफ़ियत डियण ऐ ज़ाण हासुल करण, वैंती मोकिलण, अर्ज करण, हिदायतूँ डियण, सौली सिंधीअ में ग़ालाईण.
- कम करण वारी जग़ह ते राबतो ऐ कारोबारी राबतो: ग्राहकनि सा गुफ्तगू, ऑफिस में पूछताछ, परमीशन, शिकायत पत्र, बैंक जे करमचारियुनि सा गुफ्तगू, ग्राहकनि सा ग़ालिहाइणि, बेसिक टेलीफोनिक संवादु, नंडे जशन जी रिपोर्ट.

यूनिट २: डिजिटल ऐ आधुनिक संवादु (सिंधी में):

- ईमेल ऐ मैसेज लिखण, पेशेवर ईमेल, वाट्सप ऐ समसे (Message) जा शिष्टाचार (Etiquettes), घोषणा याद डियारण वारो नोट (Reminders),
- उत्पाद ऐ प्रचार प्रसार संवादु, नंडो विज्ञापन (Small Ads), टैगलाइन ऐ स्लोगन सौलो उत्पाद वर्णन.
- पिच प्रिन्टेशन, इंटरव्यू, मार्केट रिसर्च जे लाइ संवादु.

10	Scheme of Examination and Assessment Pattern		
	Paper – 50 Marks		
	External Examination: Semester End External - 30 marks Time: 1:00 hour		
	Format of Question Paper		
	Question No	Nature of Questions	Marks
	Q1.A)	Objective Type Questions (Unit- I)	05
	Q1. B)	Attempt ANY 2 out of 4- (5 marks each) (Unit-I)	10
	Q2.A)	Objective Type Questions (Unit- II)	05
	Q2. B)	Attempt ANY 2 out of 4- (5 marks each) (Unit-II)	10
	Total		30
Internal Examination: Continuous Evolution - 20 marks			
		Total 30	
	Assessment / evaluation	Marks	
1.	Written assignment on any one of the following topics 1) Draft a formal letter and formal E_mail in Sindhi 2) Write a conversation between Customer and Shopkeeper, Nurse and patient. (Students are required to use AI assistance in the preparation of their drafts. Eg: Notion AI, Powtoon, Elicit)	15	
2.	Class Attendance and Participation	05	
Total		20	
11	REFERENCE BOOKS: 1. Sanchari Basha – By Dr. Pushpa Kodwani 2. Sindhi Pahakaa – Dr. Jetly M.K. 3. Sindhi Muhavahra – By Hardwani Lachhman 4. Sindhi Adhyat mak Shabdhkesh – By Hardwani Lachhman 5. Acho Sindhi Sikhu – By Hardwani Lachhman		

Syllabus Committee:

Sr No	Name of the Faculty	Designation and College	Signature
1.	Dr. Kajal Ramchandani	H.O.D. of Jai Hind College	
2.	Mrs. Komal Totani	Faculty In-Charge, Smt. CHM College	

Name & Signature of the BoS Chairperson: (Dr. Kajal Ramchandani) 

Name & Signature of the Dean: (Dr. Nitin Arekar) 



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- II

Title: Cocurricular Course I

**Vertical - 6
Cocurricular Course - 2 Credits**

**with effect from
Academic Year 2025-2026**



HSNC Board's
Smt. Chandibai Himathmal Mansukhani College, Ulhasnagar
(Autonomous)
Affiliated to the University of Mumbai

Bachelor of Science
(Computer Science)
(Self-Financing Course)

Semester – III

Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2025-2026

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Vertical – 1

Principles of Operating Systems	: 2 Credits
Theory of Computation	: 2 Credits
Data Structures	: 2 Credits
Computer Science Practical 3	: 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Principles of Operating Systems

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Principles of Operating Systems

Course Code: CHMCSIII1

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>The Principles of Operating Systems course offers students a comprehensive introduction to how operating systems manage hardware and software resources. Key topics include process management, memory allocation, file systems, and CPU scheduling. By exploring these core concepts, students gain a solid understanding of how computers operate at a fundamental level.</p> <p>Relevance:</p> <p>This course is crucial in bridging the gap between computer hardware and user-level applications. By understanding the role of the operating system, students gain insight into how programs execute efficiently and how critical resources such as memory and CPU are allocated and managed.</p> <p>Usefulness:</p> <p>The course provides students with hands-on knowledge applicable to configuring, using, and troubleshooting various operating systems, including Linux, Windows, and macOS. It also enhances their understanding of how applications interact with system resources—an essential skill for developers, testers, and IT professionals alike.</p> <p>Application:</p> <p>The principles of operating systems play a vital role across diverse fields such as embedded systems, mobile app development, cloud computing, and cybersecurity. Whether it's smartphones, servers, or IoT devices, a solid understanding of OS design and implementation is essential for developing and managing today's complex computing environments.</p> <p>Interest:</p> <p>This course engages students by revealing the inner mechanics of the systems they interact with daily. Through hands-on activities—such as using command-line tools, simulating scheduling algorithms, and examining file structures—it offers</p>

		<p>an intellectually stimulating and practically valuable learning experience.</p> <p>Connection with Other Courses:</p> <p>The concepts covered in this course complement and reinforce learning in related subjects such as Computer Architecture, Data Structures and Algorithms, System Programming, and Cybersecurity. Its interdisciplinary approach enhances students' comprehensive understanding of the broader computer science field.</p> <p>Demand in the Industry:</p> <p>The IT industry consistently seeks professionals with a solid understanding of operating system fundamentals. Companies involved in developing system software, embedded platforms, or managing cloud infrastructure highly value candidates with expertise in OS-level operations.</p> <p>Job Prospects:</p> <p>Mastering this subject opens up a range of career opportunities, including roles such as system administrator, kernel developer, embedded systems engineer, and DevOps engineer. In-depth knowledge of operating systems also paves the way for specialization in areas like cybersecurity, performance optimization, and system software development.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To understand the fundamental concepts and architecture of operating systems.</p> <p>CO(A) 2. To explore techniques for inter-process communication.</p> <p>CO(A) 3. To examine and analyse various CPU scheduling algorithms.</p> <p>CO(A) 4. To gain knowledge of memory management strategies and AI applications in system software.</p> <p>CO(A) 5. To study the structure, management, and implementation of file systems.</p>	

8	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <ul style="list-style-type: none"> CO1: Demonstrate proficiency in using different types of operating systems. CO2: Manage threads, processes, and synchronization effectively. CO3: Implement and analyse CPU scheduling algorithms. CO4: Understand the fundamental principles of memory management and and AI-based system software concepts. CO5: Design and develop file system structures.
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Fundamentals of Operating systems – Definition of Operating System, Operating System's role, Operating-System Operations, Functions of Operating System, Computing Environments Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, Operating-System Structure, Introduction to Artificial Intelligence in System Software. • Processes – Threads - Overview, Multicore Programming, Multithreading Models, Process Concept, Process Scheduling, Operations on Processes, Inter-process Communication. • Process Synchronization – General structure of a typical process, race condition, The Critical-Section Problem, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors. • CPU Scheduling – Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, SRTF, Priority, RR, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling), Thread Scheduling. <p>UNIT II:</p> <ul style="list-style-type: none"> • Deadlocks – System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance [(Resource Allocation Graph (RAG), Safe and unsafe states, Banker's Algorithm (brief mention or optional advanced reading)], Deadlock Detection, Recovery from Deadlock. • Memory Management – Main memory background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table, NUMA architectures (brief mention) or Page coloring and memory fragmentation. • Virtual Memory – Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Mass-Storage Structure: Overview, Disk Structure, Disk Scheduling, Disk Management.

- **File System Interface and Implementation** – File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing File-System Structure, File System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Comparative Study of Real File Systems (ext4/NTFS).

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
	Total 20	

11

REFERENCES:

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Concepts, Wiley, 9th Edition
2. Achyut S. Godbole, Atul Kahate, Operating Systems, Tata McGraw Hill
3. Naresh Chauhan, Principles of Operating Systems, Oxford Press
4. Andrew S Tanenbaum, Herbert Bos, Modern Operating Systems, 4e Fourth Edition, Pearson Education, 2016
5. Operating Systems: Three Easy Pieces by Remzi Arpaci-Dusseau Andrea Arpaci-Dusseau : <https://pages.cs.wisc.edu/~remzi/OSTEP/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Theory of Computation

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Theory of Computation

Course Code: CHMCSIII2

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>Theory of Computation examines the core principles that determine which problems are computationally solvable and the efficiency with which they can be addressed. It establishes the foundation for understanding computation through abstract models, formal languages, and rigorous logical analysis, serving as a cornerstone of theoretical computer science.</p> <p>Relevance:</p> <p>This course is highly significant as it provides the theoretical foundation for the design of algorithms, compilers, and programming languages. It enhances students' understanding of the boundaries of computation, which is essential for systematically and effectively tackling complex computational problems.</p> <p>Usefulness:</p> <p>This subject enables students to model computational problems using mathematical frameworks and abstract machines such as automata and Turing machines. These skills are crucial for optimizing algorithms, designing advanced computing systems, and developing efficient parsers for programming languages.</p> <p>Application:</p> <p>Theory of Computation has vital applications in areas such as compiler construction, artificial intelligence, natural language processing, software verification, and cryptography. It also provides essential insights into determining whether problems can be solved algorithmically and the computational resources needed to solve them.</p> <p>Interest:</p> <p>Students often find this subject intellectually engaging as it pushes them to strengthen their logical reasoning and mathematical skills. It introduces elegant problem-solving strategies and innovative approaches to modeling and</p>

		<p>classifying computational problems, fostering curiosity about the capabilities and limitations of computational machines.</p> <p>Connection with Other Courses:</p> <p>This course is closely linked to areas such as Compiler Design, Artificial Intelligence, Algorithms and Data Structures, and Programming Language Theory. It offers the essential mathematical and logical foundation needed to develop advanced concepts and tools in computer science.</p> <p>Demand in the Industry:</p> <p>Industries specializing in algorithm design, formal verification, secure systems, and artificial intelligence place a high value on the concepts covered in this course. Companies involved in automation, language processing, and software reliability increasingly seek professionals with a strong foundation in theoretical computer science.</p> <p>Job Prospects:</p> <p>Career paths include positions such as compiler engineer, language designer, research scientist, software developer, algorithm specialist, and systems programmer. This field also provides a strong foundation for pursuing advanced studies or research in computer science, particularly within theoretical and mathematical computing disciplines.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Introduce the core principles of theoretical computer science, encompassing automata theory, formal languages, and computational models.</p> <p>CO(A) 2. Build an understanding of the mathematical foundations required to describe and evaluate computational processes.</p> <p>CO(A) 3. Examine various models of computation, such as finite automata, pushdown</p>	

	<p>automata, linear bounded automata, and Turing machines.</p> <p>CO(A) 4. Analyze the decidability and computational complexity of different problems. Explore the theory of computability and the inherent limitations of algorithmic problem solving.</p> <p>CO(A) 5. Provide an overview of complexity classes (P, NP, and NP-Complete), the Halting Problem and its relevance to AI safety, and introduce Finite State Machines (FSMs) in Game AI and basic agent-based reasoning.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Explain the foundational concepts and importance of the Theory of Computation within computer science.</p> <p>CO2: Design deterministic and non-deterministic finite automata for regular languages and demonstrate their equivalence.</p> <p>CO3: Use regular expressions and formal grammars to specify and generate formal languages.</p> <p>CO4: Develop context-free grammars and analyze their ambiguity, perform simplifications, and convert them into normal forms</p> <p>CO5: Distinguish between P, NP, NP-Complete, and NP-Hard problems, recognize undecidable problems including the Halting Problem and its relevance to AI safety, and understand the role of FSMs in Game AI and basic agent-based reasoning.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Theory of Computation: Basics of Computation, Importance of Theory of Computation in Computer Science, Mathematical Foundations (Sets, Relations, Functions, Proof Techniques) • Automata Theory: Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines, DFA and NFA equivalence, Mealy and Moore Machines, Minimizing Automata. Application of Finite State Machines (FSMs) in Game AI and basic Agent-based reasoning. • Formal Languages: Defining Grammar, Derivations, Languages generated by Grammar, Chomsky Classification of Grammar and Languages, Recursive Enumerable Sets, Operations on Languages, Languages and Automata. • Regular Languages: Regular Grammar, Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties, Regular Sets and Regular Grammar. • Context Free Languages: Context-free Languages, Derivation Tree, Ambiguity of

Grammar, CFG simplification, Normal Forms, Pumping Lemma for CFG.

UNIT II:

- **Pushdown Automata:** Definitions, Acceptance by PDA, Equivalence of PDA and CFG .
- **Linear Bound Automata:** The Linear Bound Automata Model, Linear Bound Automata and Languages.
- **Turing Machines:** Turing Machine Definition, Representations, Acceptability by Turing Machines, Designing and Description of Turing Machines, Turing Machine Construction, Variants of Turing Machine, Decidability and Undecidability, The Church-Turing thesis, Universal Turing Machine, Halting Problem, Introduction to Unsolvable Problems. AI Alignment: The Halting Problem's connection to the Safety and Control of autonomous AI systems.
- **Computability and Complexity:** Time Complexity and Space Complexity, Big-O Notation, Class P and Class NP, NP-Complete and NP-Hard Problems, Polynomial Reductions, Introduction to Complexity Hierarchies

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

REFERENCES:

1. Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI, 3rd Edition
2. Introduction to Computer Theory, Daniel Cohen, Wiley, 2nd Edition
3. Introductory Theory of Computer Science, E.V. Krishnamurthy, Affiliated East West Press.
4. Theory of Computation, Kavi Mahesh, Wiley India
5. Elements of The Theory of Computation, Lewis, Papadimitriou, PHI
6. Introduction to Languages and the Theory of Computation, John E Martin, McGraw-Hill Education.
7. Introduction to Theory of Computation, Michel Sipser, Thomson.
8. Foundations of Computation, Carol Critchlow, David J. Eck, HWS Colleges, 2nd Edition, 2011, Open License (CC BY-NC-SA 3.0)
<https://math.hws.edu/FoundationsOfComputation/>
9. Introduction to Theory of Computation, Anil Maheshwari & Michiel Smid, Carleton University (2024 edition, CC BY-SA 4.0)
<https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Data Structures

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Data Structures
Course Code: CHMCSIII3

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>The Data Structures course lays the groundwork for understanding how data is organized and manipulated in computer systems. It focuses on the efficient storage, retrieval, and processing of data through a variety of fundamental and advanced data structures, forming a core component of software development and algorithm design.</p> <p>Relevance:</p> <p>Data structures are crucial to solving computational problems efficiently. The way data is structured directly influences the performance, scalability, and maintainability of software. A strong grasp of data structures enables developers to write high-performance applications and solve real-world problems with precision.</p> <p>Usefulness:</p> <p>Studying data structures helps students write efficient and scalable code. It enables better problem-solving, memory management, and algorithm optimization. This knowledge is essential for technical interviews, real-world applications, and building high-performance software systems</p> <p>Application:</p> <p>Data structures are widely used in real-world systems such as: Databases - indexing and efficient data retrieval, Compilers - syntax analysis using trees and stacks, Networking - routing using graph algorithms, AI and Machine Learning - search and decision trees, Operating Systems - job scheduling, memory management</p> <p>Interest:</p> <p>Students find data structures intellectually stimulating as it challenges them to think logically and solve problems efficiently. Building and visualizing structures like AVL Trees, heaps, or</p>

		<p>graph algorithms offers a rewarding hands-on experience that enhances both critical thinking and coding skills.</p> <p>Connection with Other Courses:</p> <p>Data Structures is foundational to many core computer science subjects. It directly supports: Algorithms – for designing efficient solutions, Operating Systems – in memory and process management, Databases – for indexing and query optimization, Programming Languages – in compiler design and syntax parsing.</p> <p>Demand in the Industry:</p> <p>Data structures are a core skill in the tech industry. Employers across software, data, and system roles expect strong proficiency in data structures for building efficient, reliable, and scalable applications. They are a major focus in technical interviews, coding tests, and system design evaluations, making them essential for career success in technology fields.</p> <p>Job Prospects:</p> <p>Mastery of data structures paves the way for careers in software engineering, data science, system architecture, web development, and cybersecurity. It lays a strong foundation for roles focused on creating efficient and scalable solutions.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Introduce the fundamental concepts of Abstract Data Types (ADTs) and various data structures to enable efficient data organization and manipulation.</p> <p>CO(A) 2. Build a solid understanding of linked structures, including singly and doubly linked lists, and demonstrate their practical applications.</p> <p>CO(A) 3. Examine stack and queue data structures, focusing on their implementation</p>	

	<p>details and real-world applications such as expression evaluation and job scheduling.</p> <p>CO(A) 4. Investigate non-linear data structures, including trees and graphs, their key operations, and their use in solving complex problems.</p> <p>CO(A) 5. Develop knowledge of priority queues, heaps, and hashing techniques, emphasizing their importance in optimizing data access and management.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Understand and implement a variety of data structures using Abstract Data Types (ADTs), recognizing their classifications and suitable applications.</p> <p>CO2: Execute key operations on linked lists—including traversal, insertion, and deletion—and apply them to practical problems such as polynomial manipulations.</p> <p>CO3: Develop stack and queue data structures using arrays and linked lists, and apply these in real-world scenarios like delimiter validation and scheduling tasks.</p> <p>CO4: Construct and traverse tree structures including Binary Trees, Binary Search Trees (BSTs), AVL Trees, Heaps, and Decision Trees, and understand their applications in searching, classification, data encoding, and Artificial Intelligence.</p> <p>CO5: Represent and analyze graphs using suitable data structures, perform BFS and DFS traversals, and apply hashing techniques for efficient searching, indexing, information retrieval, and data management in modern computing and AI applications.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Data Structures Understand the basics and definitions of data structures, including an overview of linear and non-linear types. • Abstract Data Type: Different Data Types, different types of data structures and their classifications, Introduction to ADT, Creating user-specific ADT. • Linked Structures: ADT for linked list, Advantages and Disadvantages, <ul style="list-style-type: none"> ○ Singly Linked Lists: operations such as traversing, searching, inserting (prepending), and deleting nodes. Applications like polynomial equation representation. ○ Doubly Linked Lists: ADT, benefits and drawbacks, and techniques for inserting and deleting nodes at different positions, Application like Browser History Management. <p>AI Application: Introduction to linked structures for dynamic data storage, conversion history management in chatbots, and sequence processing in intelligent systems.</p>

- **Stacks:** Stack ADT for Stack, Advantages and Disadvantages, Applications of stack like balanced delimiters, prefix to postfix notation, prefix to infix notation, infix to postfix notation, infix to prefix notation, postfix to infix notation, postfix to prefix notation.
AI Application: Use of stacks in expression evaluation, syntax parsing, Natural Language Processing (NLP), and rule-based expert systems.
- **Queues:** Queue ADT, Advantages and Disadvantages, linked representations, applications of queue like job scheduling queues, Circular Queue operations, Dequeues.
AI Application: Use of queues in AI task scheduling, request management, and state-space exploration in intelligent systems.

UNIT II:

- **Trees:** ADT for Tree Structure. Advantages and disadvantages, Binary Tree-Properties, Implementation, Traversals technique: Inorder, Preorder, Postorder, Binary Search Tree, Balanced BST, Threaded Binary Trees, AVL Trees, Applications of Tree like Huffman Coding, Self-Balancing Trees: Red-Black Trees, Splay Trees, B-Trees and B+ Trees.
AI Application: Introduction to Decision Trees as an Artificial Intelligence and Machine Learning technique for classification and decision-making problems.
- **Priority Queues and Heaps:** Priority Queue, Priority Queue ADT, Advantages and Disadvantages, Applications of heap like Least Recently Used (LRU) using heaps, Heaps, types of heaps, Heapifying the element.
AI Application: Use of priority queues and heaps in heuristic search, best-first search, intelligent task scheduling, and resource allocation in AI systems.
- **Graph:** Introduction, Graph ADT, Advantages and Disadvantages, Graph Representation using adjacency matrix and adjacency list, Graph operations like insertion and deletion of nodes, Graph Traversals using BFS & DFS, Applications of Graphs like shortest path algorithms.
AI Application: Introduction to graph-based problem solving, state-space search, knowledge graphs, recommendation systems, intelligent navigation, and route planning.
- **Hashing:** Hash Table ADT, Advantages and Disadvantages, Concept of hashing, hash table, hash functions, collision detection, collision avoidance techniques, Applications of hashing.
AI Application: Introduction to the role of hashing in Artificial Intelligence for information retrieval, indexing, similarity search, recommendation systems, and efficient data access in large-scale AI applications.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX	15

	(Each question of 5 marks, based on unit 2)	
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Introduction to Algorithm, Thomas H Cormen, PHI.
2. Data Structures And Algorithms Made Easy, Narasimha Karumanchi, 2021.
3. Fundamentals of Computer Algorithms, Sartaj Sahni and Sanguthevar Rajasekaran Ellis Horowitz, Universities Press, 2018.
4. Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Wiley, 2016.
5. Introduction to Algorithms: <https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-fall-2011/pages/lecture-notes/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Computer Science Practical 3

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Computer Science Practical 3

Course Code: CHMCSIII4

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Introduction:</p> <p>This course builds on core ideas from Operating Systems, Data Structures, and Theory of Computation. It emphasizes system-level programming and structured data handling through practical lab work. Students gain hands-on experience with concurrency, resource management, and scheduling, while also applying concepts from the Theory of Computation—such as finite automata, grammar parsing, and Turing machines—to understand how computational logic is implemented in real-world systems.</p> <p>Relevance:</p> <p>Operating Systems, Data Structures, and Theory of Computation form the backbone of computing. This practical course equips students with essential skills in data handling, system resource management, and synchronization, while also applying core ToC concepts like automata, formal languages, and computability to real-world programming and system design.</p> <p>Usefulness:</p> <p>This course establishes a strong foundation in system-level programming and abstract data management, essential for every computer science graduate. It reinforces key concepts such as scheduling, synchronization, and data abstraction through hands-on coding exercises. Students also develop practical skills in debugging, memory management, and efficient algorithm design. Additionally, the course connects to the Theory of Computation by applying principles like automata, formal languages, and computability, helping students understand how theoretical models influence real-world system design. These combined skills are crucial for both academic excellence and industry readiness.</p> <p>Application:</p> <p>The practical skills gained in this course are directly applicable to designing operating system components, file systems, and memory management tools. Data structures like trees, heaps, and graphs are essential in building</p>

compilers, databases, and network applications. Additionally, knowledge of multithreading and process synchronization prepares students to develop concurrent systems such as games, real-time simulations, and cloud platforms. By integrating principles from the Theory of Computation—such as language parsing, automata, and computational models—students also gain insight into the logic behind compilers and system-level software, which are central to modern software development.

Interest:

Students often find this course engaging due to its interactive, logic-driven approach to real-world problem solving. Concepts like circular queues for task scheduling and graphs for social network analysis spark curiosity and hands-on involvement. The course also integrates Theory of Computation principles—such as automata and formal languages—which deepen understanding of computational logic. Hands-on implementation of real-time synchronization and visual data traversals makes learning both stimulating and rewarding, encouraging creative problem-solving through coding.

Connection with Other Courses:

This practical course complements theoretical subjects such as Operating Systems, Data Structures and Algorithms, Object-Oriented Programming, and Theory of Computation. It lays the foundation for advanced fields like Distributed Systems, Artificial Intelligence, and Systems Programming. Concepts from this course, including computational models and algorithmic logic, are further explored in Database Systems, Compiler Design, and Software Engineering, making it a vital link across the curriculum.

Demand in the Industry:

The industry consistently demands professionals skilled in data structures, operating system fundamentals, and Theory of Computation. Expertise in process synchronization, memory optimization, graph traversal, and computational logic is crucial for careers in software development, cloud computing, and backend engineering. These competencies are commonly assessed in technical interviews at top product and service-based companies. The ability to apply theoretical concepts effectively in writing efficient code is highly valued across the tech sector.

		<p>Job Prospects:</p> <p>Completing this course prepares students for roles such as Software Developer, System Programmer, Backend Engineer, and Technical Consultant. It opens doors to opportunities in IT services, product development, and tech startups. With additional specialization, students can advance into fields like AI/ML, cybersecurity, and data engineering. By integrating principles from Theory of Computation alongside practical skills, the course provides a robust foundation and serves as a strong launchpad for core technical careers in the industry.</p>
2	Vertical 1	Major
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Develop hands-on skills in implementing core operating system concepts, including process management, synchronization, and memory management.</p> <p>CO(A) 2. Simulate and solve real-world problems related to operating systems, such as CPU scheduling and buffer control.</p> <p>CO(A) 3. Apply fundamental Theory of Computation concepts—like automata, formal languages, and computational models—to deepen understanding of algorithmic logic and system design.</p> <p>CO(A) 4. Strengthen understanding of data abstraction and manipulation using linked structures, trees, graphs, and hashing.</p> <p>CO(A) 5. Enhance problem-solving abilities through coding, debugging, and testing of data structure-oriented programs.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Design and implement solutions using inter-process communication methods like shared memory and message passing, and apply multithreading, synchronization, and scheduling algorithms to solve operating system problems.</p> <p>CO2: Analyse and evaluate the performance of memory and disk management techniques alongside abstract data operations.</p> <p>CO3: Apply core Theory of Computation concepts—such as automata, formal languages, and computational models—to strengthen problem-solving and system design skills.</p> <p>CO4: Construct and manipulate linear and non-linear data structures, including</p>	

	<p>stacks, queues, trees, graphs, and hash tables, and apply AI based decision tree techniques for effective algorithm development.</p> <p>CO5: Utilize practical programming skills to create efficient, real-time, and scalable system-level applications.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p>Practical 1: Threading and Single Thread Control Flow</p> <ul style="list-style-type: none"> a) Create and manage threads using POSIX threads or Java threads. b) Explore thread creation, execution order, joining, and delay handling. c) Compare execution time for sequential and threaded execution. <p>Practical 2: Multi-threading Computation and Resource Sharing</p> <ul style="list-style-type: none"> a) Generate Fibonacci sequences using multiple threads. b) Handle thread safety and synchronization with shared variables. c) Introduce concepts of thread pooling and task delegation. <p>Practical 3: Inter-Process Communication (IPC)</p> <ul style="list-style-type: none"> a) Understand shared memory and message-passing techniques. b) Implement producer-consumer problem using: <ul style="list-style-type: none"> i. Shared memory and semaphores. ii. Message queues or pipes (blocking and non-blocking). <p>Practical 4: Process Synchronization – Bounded Buffer Problem</p> <ul style="list-style-type: none"> a) Simulate the bounded buffer (producer-consumer) problem using mutexes and semaphores. b) Implement buffer control with synchronized access. c) Use circular queues for managing the shared buffer. <p>Practical 5: Readers-Writers Problem and Synchronization</p> <ul style="list-style-type: none"> a) Implement the readers-writers problem with reader and writer prioritization. b) Use semaphores to manage concurrent reader and exclusive writer access. c) Extend the solution to ensure fairness prevent deadlock <p>Practical 6: CPU Scheduling Problems</p>

- a) Simulate multiple scheduling algorithms:
 - FCFS, SJF (non-preemptive), and Round Robin.
- b) Visualize execution using Gantt charts.
- c) Compare performance metrics: waiting time, turnaround time, response time.

Practical 7: Memory Management and Page Replacement

- a) Simulate FIFO and LRU page replacement using page reference strings.
- b) Measure hit/miss ratios under different reference patterns.
- c) Extend to include frames and memory constraints.

Practical 8: Disk Scheduling and Simple File System Design

- a) Simulate Disk Scheduling Algorithms: FCFS, SSTF, C-SCAN, C-LOOK, RSS for disk head movement.
- b) Implement basic file operations (Create, Read, and Delete) and study directory structure

UNIT II:

Practical 1: Lexical Analysis and Finite Automata

- a) Perform tokenization for a given input string.
- b) Design and implement a Deterministic Finite Automaton (DFA) for a given regular language.
- c) Design and implement a Non-Deterministic Finite Automaton (NFA) for a given regular language

Practical 2: Finite State Machines:

- a) Design and simulate a Moore Machine for a given input.
- b) Design and simulate conversions between Regular Expressions and Finite Automata.

Practical 3: Context-Free Grammar and Parsing

- a) Construct a Push Down Automaton (PDA) for a given context-free grammar.
- b) Generate a Parse Tree for a given string.

Practical 4: Turing Machine

- a) Design and simulate a Turing Machine to perform a basic computation.

UNIT III:

Practical 1: Exploring Abstract Data Types (ADT) & Custom Structures

- a) Create and manipulate structures to model ADTs like

Student, Book, or Employee.

- b) Implement basic operations (create, read, update, delete) using structures.
- c) Reflect on differences between primitive and abstract data types.

Practical 2: Linked Lists and Polynomial Operations

- a) Implement singly linked lists with basic operations.
- b) Apply lists in real-life tasks (e.g., playlist, to-do list).
- c) Compare arrays vs linked lists.
- d) Represent and manipulate polynomials using linked lists.
- e) Perform polynomial addition/subtraction via list merging.

Practical 3: Working with Doubly Linked Lists

- a) Create a doubly linked list with forward and backward traversal.
- b) Implement insertion/deletion at head, tail, and specific positions.
- c) Discuss in scenarios like browser history or undo-redo features.

Practical 4: Implementing and Using Stack ADT

- a) Implement push, pop, peek using arrays or linked lists.
- b) Solve problems like delimiter matching or undo mechanism.
- c) Convert expressions from prefix to postfix and evaluate them.

Practical 5: Understanding Queues and Circular Queues

- a) Develop linear and circular queues to simulate task scheduling.
- b) Perform enqueue and dequeue with wrap-around logic.
- c) Discuss memory utilization in linear vs circular queues.

Practical 6: Tree Traversals, BST and AI-Based Decision Trees

- a) Create and analyze a Binary Search Tree (BST) using Orange Data Mining.
- b) Perform traversal-based operations using data sorting and visualization.
- c) Build a Decision Tree model in Orange and analyze its AI-based classification results.

Practical 7: Balanced Trees & Priority Queues

- Insert values and observe AVL tree rebalancing.
- Construct min-heaps or max-heaps and simulate priority queues.
- Use priority queues to manage task priorities (e.g., patient triage, job scheduling).

Practical 8: Graph Representations and Traversals

- Represent graphs using adjacency matrices and lists.
- Implement BFS and DFS to explore graph components.
- Use graphs for mapping routes or exploring social networks.

10**Scheme of Examination and Assessment Pattern**

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	12
Q.2.	Unit – II	06
Q.3.	Unit – III	12
		Total 30

Note:

- Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11**REFERENCES:**

- Silberschatz, A., Galvin, P. B., & Gagne, G. (2022). Operating system concepts (10th ed.). Wiley.
- Aho, A. V., Ullman, J. D., & Lam, M. S. (2021). Data Structures and Algorithms and Python (Adapted by R. Rao). Pearson India.
- Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). Data Structures and Algorithms in Java (6th ed.). Wiley India.
- Kanetkar, Y. (2020). Data Structures Through Python (1st ed.). BPB

Publications.

5. Algorithms, 4th Edition by Robert Sedgewick and Kevin Wayne, Addison-Wesley, 2011: <https://introc.cs.princeton.edu/java/40algorithms/>
6. Problem Solving with Algorithms and Data Structures using Python by Brad Miller and David Ranum
<https://runestone.academy/ns/books/published/pythonds/index.html>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year B. Sc.

Semester- III

Vertical – 2

Fundamentals of Operations Research : 2 Credits

**Practicals based on Fundamentals of Operations
Research : 2 Credits**

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year B. Sc.

Semester- III

Title: Fundamentals of Operations Research

Vertical - 2

Minor Course: 2 Credits

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

Title: Fundamentals of Operations Research

Course Code: CHMMTIII8

Sr. No.	Heading	Particulars
1	Description the Course:	Introduction: Operations Research (OR) is a discipline focused on the application of advanced analytical methods to support effective decision-making. This course offers a comprehensive understanding of core concepts and techniques in operations research, with emphasis on linear programming, transportation problems, and assignment problems.
2	Vertical 2	Minor
3	Type	Theory
4	Credit	2 credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	CO(A) 1. To understand the fundamental concepts and structure of Linear Programming Problems (LPP). CO(A) 2. To develop the ability to formulate and solve LPPs using appropriate methods. CO(A) 3. To analyze the concept of duality in linear programming and its managerial significance. CO(A) 4. To design and evaluate efficient transportation schedules and to apply assignment models for optimal allocation of jobs to machines.
8	Course Outcomes:	CO1: To develop the skills to formulate and solve Linear Programming Problems (LPP) for optimizing cost, time, profit, or loss. CO2: To understand and analyze the concept of duality in LPP and its significance in managerial decision-making. CO3: To design optimal transportation schedules that minimize cost and maximize efficiency and to allocate jobs to machines effectively using assignment models for optimal resource utilization. CO4: To explore advanced topics in Operations Research, including transportation problems, and assignment problems.

Syllabus

UNIT I:

- **Linear Programming:** Mathematical formulation of maximization and minimization problems; concepts of solution, feasible solution, basic feasible solution, and optimal solution. Graphical method for two-variable problems. Simplex method for problems with two or more variables, including the Big M method. Introduction to duality and its application in solving LPPs; relationship between primal and dual solutions, and the economic interpretation of dual variables.

UNIT-II:

- **Transportation Problem:** Concept and mathematical formulation; solution concepts including feasible solution and initial basic feasible solution using North West Corner Rule, Matrix Minima Method, and Vogel's Approximation Method. Optimal solution through MODI Method, including optimality testing and improvement procedures. Variants include unbalanced and maximization-type transportation problems.
- **Assignment Problem:** Concept and mathematical formulation; solution methods including Complete Enumeration and Hungarian Method. Variants include unbalanced and maximization-type problems. Introduction to the Travelling Salesman Problem (TSP) as a special case.

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes.	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
	Total	20

REFERENCES:

1. Operations Research: Kantiswaroop and Manmohan Gupta. 4th Edition; S Chand & Sons.
2. Schaum Series book in O.R. Richard Broson. 2nd edition Tata Mcgraw Hill Publishing Company Ltd.
3. Operations Research: Methods and Problems: Maurice Sasieni, Arthur Yaspan and Lawrence Friedman, (1959), John Wiley & Sons.
4. Mathematical Models in Operations Research : J K Sharma, (1989), Tata McGraw Hill Publishing Company Ltd.
5. Principles of Operations Research with Applications to Management Decisions: Harvey M. Wagner, 2nd Edition, Prentice Hall of India Ltd.
6. Operations Research: S.D.Sharma. 11th edition, Kedar Nath Ram Nath & Company.
7. Quantitative Techniques For Managerial Decisions: J.K.Sharma , (2001), MacMillan India Ltd.
8. Operations Research: Theory and Applications, J K Sharma, Trinity Press, 6th Edition , 2017
9. Introduction to Operations Research, Frederick S. Hillier, Gerald J. Lieberman, McGraw Hill Education; 11th edition, 2021
10. Operations Research, P K Gupta, S. Chand Publications, 7th Edition, 2018
11. Operations Research, P. Rama Murthy, New Age Publication, 2nd Edition
12. Operations Research: An Introduction, 10th Edition, Hamdy A. Taha, Pearson Education, 2019

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year B. Sc.

Semester- III

**Title: Practicals based on Fundamentals of
Operations Research**

Vertical - 2

Minor Course: 2 Credits

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

Title: Practicals based on Fundamentals of Operations Research

Course Code: CHMMTIII9

Sr. No.	Heading	Particulars
1	Description the Course:	Introduction: Operations Research (OR) is a scientific approach to decision-making that involves the use of advanced analytical methods. This course aims to provide a thorough understanding of fundamental OR techniques, with a primary focus on linear programming, transportation problems, and assignment problems, to support optimal and efficient decision-making.
2	Vertical 2	Minor
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<ul style="list-style-type: none">CO(A) 1. To gain a foundational understanding of the concepts and structure of Linear Programming Problems (LPP).CO(A) 2. To build the skills required to formulate and solve LPPs using suitable analytical methods and to examine the concept of duality in linear programming and interpret its managerial relevance.CO(A) 3. To develop and assess cost-effective and efficient transportation schedules.CO(A) 4. To apply assignment models for the optimal distribution of tasks or jobs to available resources.
8	Course Outcomes:	<ul style="list-style-type: none">CO1: Develop the skills to formulate and solve Linear Programming Problems (LPP) for optimizing cost, time, profit, or loss.CO2: Understand and analyze the concept of duality in LPP and its relevance to managerial decision-making.CO3: Design efficient transportation schedules to minimize cost and improve operational effectiveness.CO4: Apply assignment models for optimal job-to-machine allocation and resource utilization and explore advanced topics in Operations Research, with a focus on transportation and assignment problems.

9

Practical 1: a) Formulation and graphical solution of Linear Programming Problems (LPP).
 b) Solving minimization, maximization problems using the graphical methods

Practical 2: Application of the Simplex Method for solving minimization, maximization LPPs with standard constraints.

Practical 3: (a) Implementation of the Big-M Method for solving LPPs with artificial variables.
 (b) Application of the Two-Phase Method as an alternative to Big-M Method for artificial variable problems

Practical 4: Understanding and solving problems involving Duality.

Practical 5: a) Formulation of the Transportation Problem and obtaining the Initial Basic Feasible Solution using the **North West Corner Rule (NWCR)**.
 b) Application of the **Matrix Minima Method** to find the Initial Basic Feasible Solution. Solving Assignment Problems for optimal job allocation.

Practical 6: a) Implementation of **Vogel's Approximation Method (VAM)** for finding the Initial Basic Feasible Solution.
 b) Comparison of NWCR, Matrix Minima, and VAM in terms of initial cost effectiveness.

Practical 7: a) Computation of the **Optimal Solution using the MODI Method**.
 b) Handling **Unbalanced Transportation Problems** using balancing techniques.
 c) Solving **Maximization-type Transportation Problems** by converting them into equivalent minimization models.

Practical 8: a) Mathematical formulation and solution of **Assignment Problems** using the **Hungarian Method**.
 b) Application of the **Complete Enumeration Method** to a small-scale assignment problem for comparison.

All above practical problems solving manual and using TORA/Excel Solver.

10

Scheme of Examination and Assessment Pattern
 Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Practical 1, 2, 3 and 4	15
Q.2.	Practical 5, 6, 7 and 8	15
		Total 30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
Total		20

11

REFERENCES:

1. Operations Research: Kantiswaroop and Manmohan Gupta. 4th Edition; S Chand & Sons.
2. Schaum Series book in O.R. Richard Broson. 2nd edition Tata Mcgraw Hill Publishing Company Ltd.
3. Operations Research: Methods and Problems: Maurice Sasieni, Arthur Yaspan and Lawrence Friedman, (1959), John Wiley & Sons.
4. Mathematical Models in Operations Research : J K Sharma, (1989), Tata McGraw Hill Publishing Company Ltd.
5. Principles of Operations Research with Applications to Management Decisions: Harvey M. Wagner, 2nd Edition, Prentice Hall of India Ltd.
6. Operations Research: S.D.Sharma. 11th edition, Kedar Nath Ram Nath & Company.
7. Quantitative Techniques For Managerial Decisions: J.K.Sharma , (2001), MacMillan India Ltd.
8. Operations Research: Theory and Applications, J K Sharma, Trinity Press, 6th Edition , 2017
9. Introduction to Operations Research, Frederick S. Hillier, Gerald J. Lieberman, McGraw Hill Education; 11th edition, 2021
10. Operations Research, P K Gupta, S. Chand Publications, 7th Edition, 2018
11. Operations Research, P. Rama Murthy, New Age Publication, 2nd Edition
12. Operations Research: An Introduction, 10th Edition, Hamdy A. Taha, Pearson Education, 2019

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

**Vertical - 3
Open Elective Course: 2 Credits**

**with effect from
Academic Year 2025-2026**



HSNC Board's

Smt. Chandibai Himathmal Mansukhani College

(Autonomous)

(Affiliated to the University of Mumbai)

University College Code: 217 | JD Office: T14



Faculty of Interdisciplinary

Vertical 3: List of Open Elective Skill Based Courses for Second Year: Semester – III

Sr. No.	Nomenclature of the Paper
1	Data Analysis And Visualization Using Excel
2	2D Animation And Motion Graphics
3	Advance Tools Of AI For Economics And Education - I
4	English For Journalism And Advertising
5	Urbanization And Real Estate: Infrastructure, Technology And Urban Change
6	Tourism Marketing
7	Managing Family Wealth Through Family Office-III
8	Responsive & Modern Web Designing
9	Basics Of Nutrition -3
10	Reel Making For Media And Social Change
11	Preforming Art- Dance-3
12	Data Analysis With Excel And Power BI
13	Digital Political Strategy, AI And Public Engagement Skills
14	Psychology Of Personal Relationship-I
15	Introduction To Sociology And Digital Society
16	Mushroom Cultivation Training And Trading Level 3
17	Yogasanās: Intermediate Series
18	Perfumery Course Level 3
19	Workplace And Professional Skills
20	Beautician: Strategic Business Planning III
21	Current Trends In Fashion Designing: Financial Perspective Level 3
22	Basics Of Accounting-III
23	Digital Marketing -III
24	Advanced Trading Strategies In Stock Market



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

**Vertical - 4
Vocational Skill Course (VSC) : 2 Credits**

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Advanced JAVA

Vertical - 4

Vocational Skill Course (VSC) - 2 Credits

**with effect from
Academic Year 2025-2026**

Title: Advanced JAVA
Course Code: CHMCSIII8

Sr. No.	Heading	Particulars
1.	Description the course:	<p>Introduction:</p> <p>Java is a powerful, widely used, and platform-independent programming language based on the object-oriented programming paradigm. This course introduces learners to core and advanced Java concepts, including Collections Framework, Generics, Multithreading, Lambda Expressions, Stream API, and JSON processing. It also covers application development using Swing and JDBC, along with modern backend technologies such as Maven and Spring Boot. Students will gain hands-on experience in building RESTful APIs, performing CRUD operations, and developing database-integrated applications with advanced features.</p> <p>Relevance:</p> <p>In today's technology-driven world, Java plays a crucial role as a foundational programming language used in developing enterprise, web-based, mobile, and desktop applications. Its concepts and syntax not only support Java programming but also help in understanding other languages like C#, Kotlin, and Scala. Java remains highly relevant due to its extensive use in Android development, big data technologies such as Hadoop, and large-scale enterprise systems.</p> <p>Usefulness:</p> <p>This course builds strong Java programming and software development skills through hands-on learning of Collections, Multithreading, Lambda Expressions, Stream API, and JSON processing. It also covers GUI development using Swing with JDBC, along with Maven and Spring Boot for modern application development. Students gain practical experience in building RESTful APIs, performing CRUD operations, and developing database-integrated applications with advanced features like validation and pagination.</p> <p>Application:</p> <p>The practical component of the course enables students to develop desktop applications, interactive GUI systems, and</p>

basic web services. Through hands-on work with JDBC, Swing, multithreading, and modern frameworks, learners gain real-world experience that reflects industry-level software development practices.

Interest:

Java's simplicity, combined with its rich libraries and practical, problem-solving approach, makes learning both engaging and effective. Students enjoy developing mini-projects and interactive applications, where they can immediately see the results of their code. This hands-on experience enhances their confidence and strengthens their interest in programming.

Connection with Other Courses:

The skills gained in this course strongly support related subjects such as Data Structures, Database Management Systems, Operating Systems, Web Development, and Software Engineering. It also includes practical exposure to Spring Boot for backend development and builds a solid foundation for advanced electives like Mobile App Development and Enterprise Computing.

Demand in the Industry:

Java developers are in steady demand across industries such as finance, education, e-commerce, and healthcare. Its robustness, scalability, and extensive community support make it a preferred language for backend systems and enterprise-level applications.

Job Prospects:

After completing this course, students are well-prepared for internships and entry-level roles such as Java Developer, Backend Developer, Software Engineer, Web Application Developer, and Android App Developer. The inclusion of Spring Boot further enhances their ability to develop industry-ready backend and RESTful applications. Strong knowledge of Java and Spring Boot also supports advanced certifications, higher studies, and participation in competitive programming opportunities.

2.	Vertical 4:	VSC
3.	Type:	Practical
4.	Credits:	2 credits (1 credit = 30 Hours of Practical work in a semester)
5.	Hours Allotted:	60 Hours
6.	Marks Allotted:	50 Marks
7.	<p>Course Objectives (CO):</p> <p>CO(A) 1. Apply core Java concepts including Collections Framework, Generics, and Iteration techniques to develop efficient and structured programs.</p> <p>CO(A) 2. Design and implement multithreaded applications using Thread, Runnable, synchronization, and ExecutorService for concurrent task execution.</p> <p>CO(A) 3. Develop functional and data-driven programs using Lambda Expressions, Stream API, and JSON processing techniques for effective data handling.</p> <p>CO(A) 4. Build GUI-based applications using Swing integrated with JDBC for performing CRUD operations and database connectivity.</p> <p>CO(A) 5. Design and develop backend applications using Maven and Spring Boot, including RESTful APIs, database integration, and advanced features like validation, pagination, and sorting.</p>	
8.	<p>Course Outcomes: After successful completion of this course, students would be able to –</p> <p>CO1: Apply core Java concepts including Collections Framework, Generics, and object-oriented principles to develop efficient, reusable, and maintainable applications.</p> <p>CO2: Implement multithreading concepts such as Thread, Runnable, synchronization, and ExecutorService to design concurrent and reliable Java programs.</p> <p>CO3: Utilize functional programming concepts using Lambda Expressions, Stream API, and JSON processing for effective data handling and manipulation.</p> <p>CO4: Develop GUI-based applications using Swing integrated with JDBC to perform CRUD operations and manage database connectivity.</p> <p>CO5: Design and develop backend applications using Maven and Spring Boot, including RESTful APIs and database integration with advanced features such as validation, pagination, sorting, and structured responses.</p>	
9.	<p>Syllabus</p> <p>Unit-I Practical 1: Collections Framework and Generics:</p>	

- a) Working with Java Collections Framework and understanding List, Set, and Map interfaces.
- b) Implementing ArrayList, HashSet, and HashMap and performing basic operations.
- c) Using Iterator and enhanced for loop for iteration.
- d) Implementing sorting using Comparable and Comparator interfaces.
- e) Implementing generic classes and generic methods in Java.

Practical 2: Multithreading in Java

- a) Creating threads using Thread class.
- b) Creating threads using Runnable interface.
- c) Executing multiple tasks using a single thread.
- d) Demonstrating synchronization in multithreading.
- e) Implementing thread pool using ExecutorService with fixed-size thread pool.

Practical 3: Lambda Expressions and Stream API

- a) Implementing functional interfaces and lambda expressions in Java.
- b) Performing Stream API operations like filter(), map(), and forEach() on collections.

Practical 4: JSON Processing in Java

- a) Converting Java objects to JSON format and parsing JSON into Java objects.
- b) Performing serialization and deserialization of Java objects using JSON.
- c) Processing JSON data structures including extraction and manipulation.
- d) Converting Java Collections (List, Map) to JSON and vice versa.

Practical 5: CRUD Operations using Swing and JDBC

- a) Establishing database connection using JDBC.
- b) Designing Swing-based GUI for database operations.
- c) Performing CREATE (table creation) and INSERT operations using JDBC through Swing interface.
- d) Retrieving and displaying records using JDBC.
- e) Implementing UPDATE and DELETE operations using JDBC.

Supplementary Exercise:

- ***Familiarization with GitHub Copilot AI Tool for code generation, auto-completion, and debugging in VS Code.***

Unit-II:

Practical 6: Maven-Based Project Setup and Development Environment Configuration

- a) Installing and configuring JDK, Maven, MySQL, and IDE.

	<ul style="list-style-type: none"> b) Understanding Maven build automation and pom.xml configuration. c) Creating and executing Maven-based Java applications. <p>Practical 7: Spring Boot Fundamentals and REST API Basics</p> <ul style="list-style-type: none"> a) Creating Spring Boot applications using Spring Initializr and understanding project structure. b) Implementing Dependency Injection using @Autowired and understanding Inversion of Control (IoC). c) Creating Service classes using @Service and implementing basic business logic layer. d) Configuring application.properties and creating basic REST APIs. e) Testing Spring Boot applications. <p>Practical 8: RESTful API Development using Spring Boot</p> <ul style="list-style-type: none"> a) Developing REST Controllers and implementing GET, POST, PUT, DELETE operations. b) Handling JSON request and response using in-memory ArrayList storage. c) Testing APIs using Postman. <p>Practical 9: CRUD Operations with Database Integration using Springboot</p> <ul style="list-style-type: none"> a) Developing a management system application with CRUD operations using Spring Boot. b) Integrating MySQL/PostgreSQL database using Spring Data JPA and understanding ORM concepts. c) Creating Entity and Repository classes and performing Create, Read, Update, and Delete operations. d) Applying Bean Validation using @Valid for input data validation. e) Testing CRUD REST APIs using Postman and verifying request-response workflow. <p>Practical 10: Advanced Features in CRUD-Based Application</p> <ul style="list-style-type: none"> a) Implementing pagination, sorting, search, and filtering using Spring Data JPA. b) Adding business logic and application-specific processing. c) Creating DTOs and structured JSON response models.
10.	<p style="text-align: center;">Scheme of Examination and Assessment Pattern Paper – 50 Marks External Examination: Semester End External - 30 marks Time: 02:00 hours Format of Question Paper</p>

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11.

REFERENCES:

1. Herbert Schildt, Java The Complete Reference, Eleventh Edition, McGraw-Hill Education, 2020.
2. Bryan Basham, Kathy Sierra, Bert Bates, Head First Servlets and JSP, O'reilly (SPD), 2018
3. Ivan Bayross, Web Enabled Commercial Applications Development Using Java 2, BPB Publications
4. Java XML and JSON: Document Processing for Java SE by Jeff Friesen January 2019, Apress
5. E. Balagurusamy, Programming with Java- A Primer, Tata McGraw-Hill Education India, 2023
6. Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press, 2018
7. Joe Wigglesworth and Paula McMillan, Java Programming: Advanced Topics, Thomson Course Technology (SPD)
8. Eric Jendrock, Jennifer Ball, D Carson and others, The Java EE 5 Tutorial, Pearson Education
9. Java Parsing Collection XML JSON: Map List XML JSON Transform by Yang Hu, 2019
10. Java SE API Documentation by Oracle <https://docs.oracle.com/en/java/javase/>
11. Oracle Java EE Servlet Documentation (archived) <https://docs.oracle.com/javaee/7/tutorial/servlets.htm>
12. Introduction to Programming Using Java by Prof. David J. Eck, Hobart and William Smith Colleges, 8th Edition (2021): <https://math.hws.edu/javanotes/contents-with-subsections.html>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Vertical – 5

Ability Enhancement Course (AEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year B.Sc.

Semester- III

Vertical – 5

**Ability Enhancement Course (English)
2 Credits**

**(To be offered to Students who
opted Sindhi AEC in Sem I & II)**

**with effect from
Academic Year 2025-2026**

Title: Introduction to Communication Skills in English
Course Code: CHMBSCAECIII

Sr. No.	Heading	Particulars
1	Description the Course:	<p>Effective communication is the cornerstone of academic and professional success. This course introduces learners to foundational skills in English communication, with a focus on both oral and written competencies essential in academic, social, and workplace contexts. It aims to equip learners with the ability to read critically, write precisely, speak confidently, and listen actively. Emphasis is placed on building clarity, coherence, and conciseness in communication, along with an understanding of audience, purpose, and tone.</p> <p>The course integrates grammar reinforcement, vocabulary building, reading comprehension, and practice-oriented modules such as email etiquette, group discussion, and formal writing. Through dynamic classroom interactions and practical assessments, learners will gain confidence in using English effectively and purposefully.</p>
2	Vertical 4	Ability Enhancement Course
3	Type Teaching Methods:	Theory+ Practicum (Lecture/ Discussion/ Presentation/ Reading sessions/ Worksheets/ etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A)1: To introduce learners to the fundamentals of effective communication and its components.</p> <p>CO(A)2: To enhance learners' reading comprehension through exposure to multiple genres and contexts.</p> <p>CO(A)3: To develop grammatical accuracy and lexical resourcefulness for academic and professional communication.</p> <p>CO(A)4: To strengthen verbal and non-verbal presentation skills and promote interactive speaking abilities.</p>	

	CO(A)5: To build competence in real-world writing tasks such as email drafting, bio-data preparation, and descriptive writing.
8	Course Outcomes: Student will be able to CO-1: Understand and apply key principles of effective communication in varied contexts. CO-2: Comprehend and analyze written texts using appropriate reading strategies. CO-3: Recognize and correct common grammatical and lexical errors. CO-4: Engage in clear, confident, and context-appropriate spoken interactions. CO-5: Produce structured, coherent, and grammatically correct written content for academic and workplace use.

Syllabus

UNIT I: Foundations of English Communication

A) Core Concepts of Communication

- Principles of Effective Communication: The 7 Cs
- Verbal and Non-verbal Communication with Examples
- Cross-cultural Communication in the Globalized World
- Technology in Communication: Email, Messaging, Video Conferencing
- Listening for Detail and Intent: Barriers to Listening and Strategies

B) Reading Comprehension

- Understanding the Main Idea and Supporting Details
- Interpreting Tone, Purpose, and Bias
- Using Context Clues for Vocabulary Building
- Reading Visual Texts: Graphs, Charts, and Infographics
Sample readings will include excerpts from news articles, reports, editorials, and educational essays (200–250 words).

C) Grammar and Vocabulary

- Subject-Verb Agreement
- Sentence Structures
- Punctuation and Capitalization
- Commonly Confused Words
- Editing and Proofreading Practice

A remedial and functional approach will be followed with contextual exercises.

UNIT II: Applied Communication Skills

A) Speaking and Listening Skills

- Introducing Oneself in Academic/Professional Settings
- Participating in Group Discussions and Expressing Opinions
- Delivering a Short Speech (2–3 minutes) on Familiar Topics
- Understanding and Responding to Instructions
- Listening Comprehension Practice through Audio/Video Clips

B) Functional Writing Skills

- Formal Email Writing with Subject and Tone Sensitivity
- Descriptive Paragraph Writing (People, Places, Processes)
- Bio-data and Resume Writing
- Drafting Job Applications (Solicited and Unsolicited)
- Writing a Statement of Purpose

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

Question No	Nature of Questions	Marks
Q. 1	Short Notes (Attempt any 3 out of 5) - Unit 1 OR Essay-Type Question (Attempt any 1 out of 2)- Unit 1	15
Q. 2	Short Notes (Attempt any 3 out of 5) - Unit 2 OR Essay-Type Question (Attempt any 1 out of 2)- Unit 2	15
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Students are required to draft a job application letter along with a resume using the following AI assistance: Canva Resume Builder, Resume.oi, Zety, Novopresume, Rezi etc OR Draft an SoP with the help of the following AI assistance: Quillbot, Yocket, Writesonic, Jasper AI	15
2.		05
	Total	20

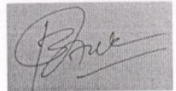
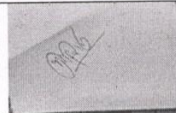
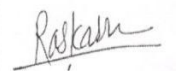
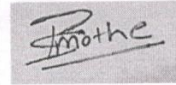
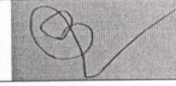
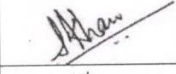
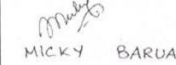

11

REFERENCES:

1. Adler, Ronald B., et al. *Understanding Human Communication*. 15th ed., Oxford UP, 2021.
2. Bailey, Stephen. *Academic Writing: A Handbook for International Students*. 5th ed., Routledge, 2018.
3. Barrass, Robert. *Students Must Write: A Guide to Better Writing in Coursework and Examinations*. Routledge, 2005.
4. Brown, Gillian, and George Yule. *Teaching the Spoken Language: An Approach Based on the Analysis of Conversational English*. Cambridge UP, 1983.
5. Carnegie, Dale. *The Quick and Easy Way to Effective Speaking*. Pocket Books, 2006.
6. Chaney, Lillian, and Jeanette Martin. *Intercultural Business Communication*. 6th ed., Pearson, 2014.
7. Cullen, Pauline, et al. *English Grammar in Use Supplementary Exercises*. Cambridge UP, 2019.
8. Eastwood, John. *Oxford Guide to English Grammar*. Oxford UP, 2005.

9. Gerson, Sharon J., and Steven M. Gerson. *Technical Communication: Process and Product*. 9th ed., Pearson, 2021.
10. Hewings, Martin. *Advanced Grammar in Use: A Self-study Reference and Practice Book for Advanced Learners of English*. 3rd ed., Cambridge UP, 2013.
11. Jones, Leo. *Functions of English: Communication Practice in English*. Cambridge UP, 1981.
12. Kumar, Sanjay, and Pushp Lata. *Communication Skills*. 2nd ed., Oxford UP, 2018.
13. Lynch, Tony. *Listening in Language Learning*. Longman, 1988.
14. McCarthy, Michael, and Felicity O'Dell. *Academic Vocabulary in Use*. Cambridge UP, 2008.
15. Nordquist, Richard. *The Essentials of English Grammar*. McGraw-Hill Education, 2016.
16. Quirk, Randolph, et al. *A Comprehensive Grammar of the English Language*. Longman, 1985.
17. Seely, John. *The Oxford Guide to Writing and Speaking*. Oxford UP, 2005.
18. Straus, Jane, et al. *The Blue Book of Grammar and Punctuation*. 12th ed., Jossey-Bass, 2021.
19. Wallace, Catherine. *Reading*. Oxford UP, 1992.
20. Zinsser, William. *On Writing Well: The Classic Guide to Writing Nonfiction*. Harper Perennial, 2016.

Syllabus Committee:

Sr. No	Name of the Faculty	Designation and College	Signature
1.	Prof. (Dr.) Kailas Aute	Professor & Head, Dept. of English, Smt. CHM College	
2.	Prof. (Dr.) B. R. Hiramani,	(VC Nominee, University of Mumbai) Pancham Khemraj College, Sawantwadi	
3.	Prof. (Dr.) Vikas Raskar	(Subject Expert outside University) Hutatma Rajguru Mahavidyalay, Rajguru Nagar, Khed, (Affiliated to Savitribai Phule University)	
4.	Prof. (Dr.) Prashant Mothe	(Subject Expert outside University) Aadarsh Mahavidyalay, Umerga, Dharashiv, (Affiliated to Dr. Baba Saheb Ambedkar Marathwada University)	
5.	Mr. Ananda Pandhare	Asst. Professor, Dept. of English, Smt. CHM College	
6.	Ms. Sana Khan	Asst. Professor, Dept. of English, Smt. CHM College	
7.	Dr. Micky Barua	Faculty Vidyalandkar Institute of technology, Alumni Member	 MICKY BARUA
8.	Ms. Sofy Verghese	Accenture, Industry Representative	

Name & Signature of the Ad-hoc BoS Chairperson: Prof. (Dr.) Kailas Aute



Name & Signature of the Dean: Prof. (Dr.) Nitin Arekar



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B.A
(Hindi)**

Semester – III

Title : हिंदी भाषा : कौशल के आधार

**Vertical - 5
Ability Enhancement Course 2 Credits**

**with effect from
Academic Year 2025-2026**

Title : हिंदी भाषा : कौशल के आधार

Course Code : CHMAECHINIII

Sr.No.	Heading	Particulars
1.	Description of the Course :	विद्यार्थियों के लिए हिंदी एक सामान्य भाषा होने के साथ विशेष भाषा तब बन जाती है जब वह हिंदी के माध्यम से अपने कौशल में अभिवृद्धि करें, हिंदी के माध्यम से रोजगार के कई अवसरों को प्राप्त करें, इस दृष्टि से पाठ्यक्रम अत्यंत लाभवर्धक और उपयोगी सिद्ध होगा, हिंदी भाषा में कौशल विकास की असीम संभावनाएं हैं और कौशल के विभिन्न आयाम जुड़े हुए हैं जो अलग – अलग दिशाओं में देखे जा सकते हैं, पाठ्यक्रम विद्यार्थियों में लेखन, वाचन कौशल की अभिवृद्धि करने के साथ रोजगारपरक अवसर प्रदान करता है।
2.	Vertical : 5	AEC
3.	Type : Teaching Method	Theory + Practicum Lecture / Discussion / Presentation / Self Study, etc.
4.	Credit :	2 Credits
5.	Hours Allotted :	30 Hours
6.	Marks Allotted :	50 Marks
7.	Course Objectives : CO(A)1 : विद्यार्थियों को लेखन, वाचन कौशल का ज्ञान देना एवं रोजगार के अवसरों से जोड़ना। CO(A)2 : विद्यार्थियों को लेखन, वाचन कौशल से परिचय करते हुए अभिव्यक्ति की शैलियों का विकास करना। CO(A)3 : विद्यार्थियों को भाषण कला के विविध रूपों को समझाना, मौलिकता में अभिवृद्धि लाना एवं विशेषज्ञता दिलाना। CO(A)4 : विद्यार्थियों को श्रवण कौशल की विशेषताओं से परिचय कराते हुए श्रवण कौशल के लाभों से अवगत कराना।	

8.	<p>Course Outcomes :</p> <p>CO1 : विद्यार्थियों का लेखन, वाचन कौशल के ज्ञान प्राप्ति के साथ मौलिक अभिव्यक्ति में बदलाव आएगा ।</p> <p>CO2 : विद्यार्थियों का लेखन, वाचन कौशल द्वारा मानसिक विकास होगा, पठन शक्ति, शैली का विकास होगा ।</p> <p>CO3 : विद्यार्थियों को लेखन, भाषण कौशल से भषिक – शक्ति, शैलियों का संवर्धन होगा विशेषज्ञता आएगी ।</p> <p>CO4 : विद्यार्थियों को लेखन, वाचन, श्रावण, भाषण कौशल की विशेषताओं और उपयोगिता का ज्ञान प्राप्त होगा ।</p>
9.	Syllabus
	UNIT I : हिंदी भाषा कौशल के आधार
	<p>1.1 लेखन कौशल का अर्थ एवं स्वरूप लेखन कौशल की उपयोगिता एवं महत्व</p> <p>1.2 लेखन कौशल की विधियाँ एवं विशेषताएँ</p> <p>1.3 वाचन कौशल का अर्थ, स्वरूप एवं विशेषताएँ</p> <p>1.4 वाचन कौशल की उपयोगिता एवं विधियाँ</p>
	UNIT II : हिंदी भाषा कौशल के आधार
	<p>2.1 भाषण कौशल का अर्थ एवं स्वरूप</p> <p>2.2 भाषण कौशल का महत्व एवं उपयोगिता</p> <p>2.3 भाषण कौशल की विधियाँ एवं विशेषताएँ</p> <p>2.4 श्रवण कौशल का अर्थ, स्वरूप एवं विशेषताएँ</p> <p>2.5 श्रवण कौशल का महत्व एवं उपयोगिता</p>

10.

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination : Semester End External – 30 Marks Time : 1:00 Hour

Format of Question Paper

All Questions are Compulsory

मूल्यांकन प्रारूप	इकाई	अंक
बाह्य मूल्यांकन		
प्रश्न 1 : चार प्रश्नों में से किन्हीं दो प्रश्नों के उत्तर लिखिए ।	इकाई 1	15
प्रश्न 2 : चार प्रश्नों में से किन्हीं दो प्रश्नों के उत्तर लिखिए ।	इकाई 2	15 □
कुल अंक		30

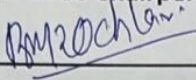
मूल्यांकन प्रारूप	अंक
आंतरिक मूल्यांकन	
<ul style="list-style-type: none"> ● AI Writing Tools की सहायता से हिंदी लेखन कौशल का अभ्यास, भाषा-संपादन, व्याकरण सुधार, सारांश लेखन एवं रचनात्मक लेखन करना। ● AI की सहायता से दिए गए विषयों पर भाषण, लेख, संवाद एवं लघु-प्रस्तुति तैयार करना तथा भाषा, शैली एवं प्रभावशीलता का विश्लेषण करना। ● AI Voice Tools का उपयोग करके हिंदी वाचन, भाषण, उच्चारण, स्वर, गति एवं प्रवाह का अभ्यास करना तथा AI आधारित Feedback प्राप्त करना। ● AI Speech-to-Text एवं Text-to-Speech Tools की सहायता से श्रवण एवं वाचन कौशल विकसित करना तथा उच्चारण की शुद्धता का अभ्यास करना। ● AI की सहायता से Reading Comprehension, प्रश्नोत्तर, शब्दार्थ, शब्दावली (Vocabulary) एवं भाषा-अभ्यास गतिविधियाँ तैयार करना। ● AI आधारित Mock Interview, Group Discussion तथा Public Speaking गतिविधियों के माध्यम से भाषण एवं संप्रेषण कौशल विकसित करना। ● AI की सहायता से हिंदी Podcast, Audio Narration एवं Listening Exercises तैयार करना तथा श्रवण कौशल का मूल्यांकन करना। 	20
कुल अंक	20

11.	<p>संदर्भ ग्रंथ सूची –</p> <ol style="list-style-type: none">1. हिंदी भाषा शिक्षण के विविध आयाम – प्राध्यापक डॉ. राठौर, किनले एडिशन2. अभिनव पत्र लेखन – डॉ. अनिल सिंह3. हिंदी के व्यावहारिक रूप – डॉ. संतोष मोटवानी, परिदृश्य प्रकाशन, मुंबई4. हिंदी भाषा लेखन कौशल – गुलीबाबा पब्लिकेशन प्राइवेट लिमिटेड
-----	--

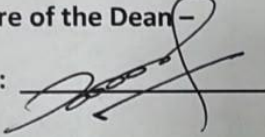
Bos in Hindi :

Sr No	Name of the Faculty	Designation and College
1.	Dr. Bhavna M.Rochlani	I/C HOD Asst. Professor CHM College Ulhasnagar
2.	Dr. Ajeet Kumar Rai	Associate Professor KC College Mumbai
3.	Dr. Santosh Motwani	Associate Professor RKT College Ulhasnagar

Name & Signature of the Ad-hoc BoS Chairperson -

Dr. Bhavna M. Rochlani : 

Name & Signature of the Dean -

Dr. Nitin Arekar : 



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year BA/BCom/BSc/SFC
(Marathi)**

Semester- III

**Vertical -5
Ability Enhancement Course (AEC) -2 Credits**

**with effect from
Academic Year 2026-2027**

Title: लेखन कौशल्ये – १ (कार्यालयीन लेखनव्यवहार आणि पत्रव्यवहार)

COURSE CODE: CHMAECMARIII

Sr. No.	Heading	Particulars
1	Description the Course:	(कार्यालयीन लेखनव्यवहार आणि पत्रव्यवहार) लेखन ओळख ते लेखन कौशल्य हा बराच मोठा प्रवास आहे. वाचन आणि लेखनाच्या सरावाने, लेखन कौशल्य विकसित करता येते. बहुतेक वेळा आपण मिळवलेले ज्ञान हे लिखित स्वरूपात मांडावे लागते. त्यासाठी आपण लेखन कौशल्याचे योग्य उपयोजन करतो. लेखने म्हणजे मजकूर तंतोतंत उतरवणे नव्हे. एखादे निवेदन, वृत्त, निबंध, पुस्तकाची टिपणे, अर्ज यांसाठी लेखन आवश्यक असते. कार्यालयीन पत्रव्यवहार, कार्यवृत्ते, नोंदी, जाहिरात, टिप्पणी ही सर्व उपयोजित लेखन कौशल्ये आहेत. कार्यालयीन पत्रव्यवहार करणे हे एक वेगळ्या प्रकारचे कौशल्य आहे. त्यातील काही उपयोजन कौशल्यांचा विचार या अभ्यासपत्रिकेत अपेक्षित आहे. कार्यालयीन लेखन व्यवहार आणि पत्रव्यवहार या अभ्यासपत्रिकेत शिकविला जाईल.
2	Vertical 5	Ability Enhancement Course
3	Type	Theory
4	Credit	2 Credits (1 Credit = 15 Hours for Theory or 30 Hours of Practical Work in a Semester)
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	CO(A) 1: कार्यालयीन लेखन व्यवहार स्वरूप समजावून सांगणे. CO(A) 2: कार्यालयीन पत्रव्यवहाराचे स्वरूप समजावून सांगणे, CO(A) 3: प्रभावी कार्यालयीन लेखनासाठी आवश्यक असणाऱ्या क्षमता आणि तंत्रांचा परिचय करून देणे,
8	Course Outcomes:	प्रस्तुत अभ्यासक्रम शिकल्यानंतर: CO1: विद्यार्थ्यांना कार्यालयीन लेखन व्यवहाराचे स्वरूप समजेल, CO2: विद्यार्थ्यांना कार्यालयीन पत्रव्यवहाराचे स्वरूप समजेल. CO3: प्रभावी कार्यालयीन लेखनासाठी आवश्यक असणाऱ्या तंत्रांचा विद्यार्थ्यांना परिचय होईल.

Syllabus

9

UNIT I कार्यालयीन लेखनव्यवहार

१. जाहीर निवेदन आणि माहितीपत्रक
२. इतिवृत्त लेखन
३. टिप्पणी लेखन

(६० मिनिटांच्या १५ तासिका, श्रेयांकन १)

(सूचना : विद्यार्थ्यांमध्ये उपरोक्त कार्यालयीन लेखन व्यवहार व पत्रव्यवहार करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित होतील या दृष्टीने शिक्षकांनी सराव करून घ्यावा.)

UNIT II: कार्यालयीन पत्रव्यवहार

१. कार्यालयीन/प्रशासनिक पत्र
२. नोकरीसाठी अर्जलेखन
३. पत्रात्मक लेखन: नवी रूपे (शुभेच्छा, निमंत्रण)

(६० मिनिटांच्या १५ तासिका, श्रेयांकन-१)

(सूचना : विद्यार्थ्यांमध्ये उपरोक्त कार्यालयीन लेखन व्यवहार व पत्रव्यवहार करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित होतील या दृष्टीने शिक्षकांनी सराव करून घ्यावा.)

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00

hours

Format of Question Paper

All questions are compulsory:

Q. No	Nature of Questions	Marks
Q1	Essay type question on Module 1	10
Q2	Essay type question on Module 2	10
Q6	MCQs 15 out of 20, 10 MCQs on each module	10
		Total 30

Internal Examination: Continuous Evaluation - 20 marks

	Project and presentation / Viva	Marks
1.	<ul style="list-style-type: none">AI च्या साहाय्याने जाहीर निवेदन आणि माहितीपत्रक तयार करणे. त्यामध्ये शीर्षक, उद्दिष्ट, कार्यक्रमाचे वेळापत्रक, संपर्क इत्यादींची आकर्षक मांडणी करणे.AI साधने वापरून सभेचे Audio/Video नुसार इतिवृत्त तयार करणे.	20

		<p>त्यानंतर विद्यार्थ्यांनी त्यात आवश्यक ती सुधारणा करणे.</p> <ul style="list-style-type: none"> ● AI चा वापर करून कार्यालयीन टिप्पणी तयार करणे. त्यातील भाषा, रचना, औपचारिकता इत्यादींचे परीक्षण करून सुधारित टिप्पणी तयार करणे. ● AI साधनांच्या माध्यमातून प्रशासनिक पत्रांचे विविध नमुनारूप तयार करणे. भाषेची औपचारिकता तपासणे. ● AI साधनांच्या साहाय्याने नोकरीसाठी अर्ज तयार करणे. दिलेल्या जाहिरातीवर आधारित Cover Letter तयार करणे. ● विविध प्रसंगांसाठी AI साधनांच्या आधारे निमंत्रणपत्र व शुभेच्छापत्र तयार करणे. <p>AI साधने: ChatGPT, Google Gemini, Claude, Perplexity AI, NotebookLM, Canva, CapCut, InVideo, Grammarly, QuillBot, Whisper, ElevenLabs</p>		
				Total 20
11	<p>संदर्भ ग्रंथ (Reference Books) :</p> <ol style="list-style-type: none"> १. प्रशासनिक लेखन, भाषा संचालनालय, महाराष्ट्र शासन, मुंबई, १९६६ २. भाषिक सर्जन आणि उपयोजन, राजन गवस, अरुण शिंदे, गोमटेश्वर पाटील, दर्या प्रकाशन, पुणे, २०१२ ३. परब प्रकाश, व्यावहारिक मराठी, मिथुन प्रकाशन, डोंबिवली पूर्व, मुंबई, १९८९ ४. नाईक सदानंद, राजभाषा मराठी, व्यावहारिक मराठी, प्रका-नागरी सेवा प्रबोधिनी, मुंबई, २००२ ५. तावरे स्नेहल (संपा.), व्यावहारिक मराठी, स्नेहवर्धन प्रकाशन, पुणे, चौथी आवृत्ती, २०११ ६. केतकी मोडक, संतोष शेणई, सुजाता शेणई (संपा.), उपयोजित मराठी, पद्मगंधा प्रकाशन, २०१२ ७. नसीराबादकर ल. रा., व्यावहारिक मराठी, भाषा विकास संशोधन संस्था, कोल्हापूर २०२३ 			

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Vertical – 6

**Field Project (FP) : 2 Credits
Co-Curricular (CC) : 2 Credits**

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- III

Title: Field Project

**Vertical - 6
Field Project - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Field Project
Course Code: CHMCSIII9

Sr. No.	Heading	Particulars
1	Description the Course:	The Field Project course, introduced under CHM Autonomy in alignment with NEP 2020, aims to connect theoretical computer science concepts with real-world technological applications. It offers students practical exposure to industry-relevant environments through field visits, case studies, system analysis, and hands-on project development. By engaging directly with technology-driven challenges such as software development, data analysis, cybersecurity, and automation, students strengthen their technical, analytical, and problem-solving abilities. The course also promotes teamwork, innovation, and ethical digital practices. Ultimately, it prepares learners for employability in the evolving tech sector and encourages them to contribute effectively to digital transformation and nation-building.
2	Vertical 6	Field Project
3	Type	Field Work
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. Explore real-world issues by conducting surveys, interviews, and field-based observations. CO(A) 2. Collaborate with users, community groups, or institutions to gain first-hand insights into their needs and challenges. CO(A) 3. Collect and systematically analyse primary data to extract relevant patterns and conclusions. CO(A) 4. Develop actionable solutions or recommendations informed by the field research outcomes. CO(A) 5. Design and prototype digital products—ranging from mobile apps and web platforms to interactive tools and game development projects—that directly address the identified challenges. CO(A) 6. Implement community-oriented initiatives, including awareness drives, skill-building workshops, or educational programs, to address the gaps highlighted through the study.	
8	Course Outcomes: After successful completion of this course, students would be able to - CO1: Integrate theoretical computer science concepts with real-world applications by	

	<p>examining practical technological problems and user needs.</p> <p>CO2: Apply critical thinking, collaborative skills, and technical decision-making during field-based tasks such as requirement analysis, prototyping, or solution design.</p> <p>CO3: Evaluate the role of empirical research and hands-on project experience in driving digital innovation and supporting societal and national development.</p>						
<p>9</p>	<p style="text-align: center;">Guidelines for Field Project</p> <p>The following general guidelines apply to the conduct of the Field Project for Semester III.</p> <p><u>Head of the Department (HOD)/ Field Project Co-ordinator</u></p> <ol style="list-style-type: none"> 1. Ensure that the Field Project (FP) aligns with departmental and academic goals, integrating NEP 2020 principles within the computer science syllabus framework. 2. Appoint faculty members as project in-charges for student groups working on domains such as software development, data analysis, networking, cybersecurity, artificial intelligence, and game development. 3. Conduct an orientation session for Field Project Supervisors and finalize the project timeline. 4. Provide necessary support to students throughout the project duration. <p><u>FP Supervisor:</u></p> <ol style="list-style-type: none"> 1. To give guidelines and objectives for the field project. 2. Oversee student progress and offer guidance as needed. <p><u>Project (Dissertation) Report:</u></p> <p>Students are required to submit a report of the field project at the end of the semester in following suggested format:</p> <ul style="list-style-type: none"> • The main text should be typed in Times New Roman, 12 point font, single-spaced, 1.5 line spacing. • Headings should be center-aligned, written in 20 point Times New Roman, bold, and in ALL CAPS. • The field project report should include student details and the signature of the Field Project Incharge. Report must be at least 10 pages long. The field visit report should capture observations, system workflows, software interactions, and key insights from practical on-site activities. It should present a complete account of the project, including the problem statement, literature review, methodology, data collection and analysis, implementation process, results, and conclusions with recommendations 						
<p>10</p>	<p style="text-align: center;">Scheme of Examination and Assessment Pattern Paper – 50 Marks</p> <p style="text-align: center;">External Examination – 30 Marks</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Component</th> <th style="text-align: center;">Marks</th> </tr> </thead> <tbody> <tr> <td>Quality of Project (Design, Creativity, Originality)</td> <td style="text-align: center;">05</td> </tr> <tr> <td>Working of Project (Functionality, Error-Free Execution, Features)</td> <td style="text-align: center;">15</td> </tr> </tbody> </table>	Component	Marks	Quality of Project (Design, Creativity, Originality)	05	Working of Project (Functionality, Error-Free Execution, Features)	15
Component	Marks						
Quality of Project (Design, Creativity, Originality)	05						
Working of Project (Functionality, Error-Free Execution, Features)	15						

Project Presentation (Demonstration, Clarity, Confidence)	10
Total	30

Internal Evaluation by Guide – 20 Marks

Component	Marks
Report Quality	10
Viva / Understanding of Game Logic	10
Total	20

11

Appendix

Name of the College /Department / Institute

CERTIFICATE

This is to certify that _____ Roll No. ____ of S.Y.B.Sc. (Computer Science) (semester III) class, has completed the Field Project _____ in the area of _____ to be submitted in the partial fulfilment for the award of Bachelor of Science in Computer Science during the Academic Year _____.

Field Project Incharge

Head of the Department

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- III

Title: Cocurricular Course I

**Vertical - 6
Cocurricular Course - 2 Credits**

**with effect from
Academic Year 2025-2026**



HSNC Board's
Smt. Chandibai Himathmal Mansukhani College, Ulhasnagar
(Autonomous)
Affiliated to the University of Mumbai

Bachelor of Science
(Computer Science)
(Self-Financing Course)

Semester – IV

Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2025-2026

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 1

Computer Networks	: 2 Credits
Software Engineering	: 2 Credits
IoT Technologies	: 2 Credits
Computer Science Practical 4	: 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Title: Computer Networks

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Computer Networks

Course Code: CHMCSIV1

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Computer Networks course offers a solid foundation in understanding how computers and devices communicate across different types of networks. It introduces essential networking models, including OSI and TCP/IP, examines the functions of each layer—physical, data link, network, transport, and application—and explores key protocols such as Ethernet, IP, TCP, HTTP, and DNS. The course also highlights modern advancements like IPv6 and practices of secure communication.</p> <p>Relevance:</p> <p>In today’s connected world, a strong understanding of computer networks is essential for computer science graduates. This course addresses the growing demand for professionals capable of designing, troubleshooting, and managing complex networked systems, including those in cloud computing, data centers, mobile communications, and IoT environments.</p> <p>Usefulness:</p> <p>This course is valuable for learners as it provides both theoretical foundations and practical insights into how data travels across networks. The knowledge gained is essential not only for networking roles but also for careers in software development, cybersecurity, systems administration, and IT infrastructure management.</p> <p>Application:</p> <p>The concepts taught in this course can be directly applied to configuring LANs and WANs, managing IP addresses, analyzing network traffic, securing communications, and developing network-based applications. Students will gain the skills to design and implement communication systems that are efficient, scalable, and secure.</p> <p>Interest:</p>

		<p>Students are often drawn to this course because of its experiential learning approach and its clear link to modern technologies such as the internet, smartphones, social media, and streaming platforms. Through simulations and practical case studies, they gain a deeper understanding of how networking theories function in real-world network applications.</p> <p>Connection with Other Courses:</p> <p>This course is closely linked with subjects like Operating Systems, Database Management Systems, Web Programming, Cloud Computing, and Cybersecurity. A solid understanding of networking is crucial for comprehending distributed systems, server request handling, and maintaining secure communications.</p> <p>Demand in the Industry:</p> <p>The industry continues to have a strong demand for professionals skilled in networking principles and network management. Networking expertise is highly valued in roles related to cloud platforms (such as AWS and Azure), DevOps, and IT support. A solid understanding of network protocols and architectures is particularly crucial for careers in network engineering, cybersecurity, and system integration.</p> <p>Job Prospects:</p> <p>Completing the Computer Networks course prepares students for roles such as network administrator, support technician, and system engineer. It provides a strong foundation for careers in cybersecurity, cloud computing, and IT infrastructure, while also supporting industry certifications and opportunities in technology-driven sectors.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To Introduce the fundamental concepts of computer networks, including network types, architectures, and models such as OSI and TCP/IP.</p> <p>CO(A) 2. To describe data transmission techniques, bandwidth utilization strategies, switching mechanisms, and various transmission media employed in modern networking systems.</p>	

	<p>CO(A) 3. Describe data link layer functions such as MAC addressing, error detection and correction, and LAN protocols.</p> <p>CO(A) 4. Demonstrate understanding of network layer operations, including IP addressing, routing, and protocol analysis for IPv4 and IPv6.</p> <p>CO(A) 5. Introduce transport and application layer protocols, their applications such as web communication, email, and DNS, familiarize students with modern network trends including IPv6 and secure communications, and provide an overview of AI-based network traffic prediction for efficient bandwidth utilization.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <ul style="list-style-type: none"> CO1: Describe key concepts of network architectures, models, and the layered structure of data communication systems. CO2: Examine the operations of the physical and data link layers, focusing on signal transmission, media types, error detection, and MAC protocols. CO3: Explain how switching, multiplexing, and addressing (IPv4/IPv6) techniques contribute to efficient data transfer and routing. CO4: Differentiate between transport protocols such as TCP, UDP, and SCTP, and apply them to various real-time networking applications. CO5: Analyze application layer protocols (HTTP, FTP, Email, DNS, etc.), explore advancements such as 5G, satellite, and secure communication networks, and understand AI-based network traffic prediction for efficient bandwidth utilization.
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Networking: Networking standards and administrations, types of networks – LAN, MAN, WAN and Network topologies. • Network Models: OSI reference model, TCP/IP protocol suite, and their comparative study. • Physical Layer Concepts: Overview of data and signals, Signal impairments: attenuation, noise, distortion, characteristics of periodic analog and digital signals, Digital transmission: line coding, analog-to-digital conversion, Transmission modes: bandwidth, throughput, latency along with various transmission modes: simplex, half-duplex, full-duplex. • Bandwidth Utilization: Multiplexing: FDM, TDM, WDM, and spread spectrum techniques for efficient use of bandwidth. • Transmission Media: Guided Media (Twisted Pair, Coaxial, Fiber Optics) & Unguided Media (Radio, Microwave, Infrared)– characteristics and applications. • Switching Techniques: Circuit switching and packet switching concepts. • Data Link Layer: Link layer addressing (MAC), design issues, and reliable communication, framing concepts, error detection and correction using CRC, checksums, data link control protocols (HDLC, Stop-and-Wait, Go-Back-N, PPP), Introduction to MAC: CSMA/CD, CSMA/CA.

- **Case Study:** AI-Based Network Traffic Prediction for Efficient Bandwidth Utilization.

UNIT II:

Wired LANs – Ethernet: Ethernet protocol, standard Ethernet, Fast Ethernet, Gigabit Ethernet, and 10 Gigabit Ethernet.

- **Wireless Networks:** IEEE 802.11 ,Bluetooth, , LoRaWAN, and Zigbee (IEEE 802.15.4)
- **Network Layer Concepts:** Network layer services, packet switching, performance analysis, and IPv4 addressing.
- **Network Layer Protocols:** Internet Protocol (IP), ICMPv4, and Mobile IP.
- **Secure Communication Protocols:** HTTPS, TLS/SSL, IPsec, and VPN operations.
- **Routing:** Unicast routing, routing algorithms, and routing protocols(RIP & OSPF).
- **Next-Generation Internet Protocols:** IPv6 addressing, IPv6 protocol, ICMPv6, and transition from IPv4 to IPv6.
- **Transport Layer:** Services and protocols – UDP, TCP: Features, 3-way handshake
- **Application Layer:** Client-server programming, and standard protocols – WWW, HTTP/HTTPS, FTP, Email (SMTP/IMAP/POP3), SSH, and DNS.
- **Case Study:** AI-Based Intrusion Detection and Intelligent Routing in Computer Networks.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q.No.	Structure of the Questions	Marks
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
	1. Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
	2. Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
	Total 20	
11	REFERENCES: <ol style="list-style-type: none"> 1. Data Communications and Networking, Behrouz A. Forouzan, Fifth Edition, TMH, 2018. 2. Computer Network, Andrew S. Tanenbaum, David J. Wetherall, Fifth Edition, Pearson Education, 2018. 3. Computer Network, Bhushan Trivedi, Oxford University Press, 2016 4. Data and Computer Communication, William Stallings, PHI, 2017 5. https://learn.microsoft.com/en-us/training/modules/network-fundamentals/ 6. https://www.thethingsnetwork.org/docs/lorawan/ 	

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Title: Software Engineering

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Software Engineering

Course Code: CHMCSIV2

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>Software Engineering lays the groundwork for systematic software development. It familiarizes learners with the complete software lifecycle—covering planning, analysis, design, development, testing, and maintenance. The course emphasizes a structured and disciplined approach to building software that aligns with user requirements and delivers dependable performance in real-world environments.</p> <p>Relevance:</p> <p>In today’s digital era, software plays a critical role across nearly all sectors. This subject is essential for preparing students to manage growing software complexity and to appreciate the significance of process-driven development and quality standards.</p> <p>Usefulness:</p> <p>The course provides students with practical tools and methodologies for software planning, estimation, design, and testing. It also develops analytical thinking and technical communication skills, enabling students to manage real-world software projects effectively and efficiently.</p> <p>Application:</p> <p>Software engineering principles are applied across diverse domains such as banking, healthcare, education, and mobile application development. The skills acquired in this course enable students to manage complete projects while ensuring product scalability and user satisfaction.</p> <p>Interest:</p> <p>The course features interactive topics such as UML modelling, Agile methodologies, risk management, and testing strategies. Students actively apply these concepts through diagrams, real-world case studies, and collaborative project exercises.</p>

		<p>Connection with Other Courses:</p> <p>This subject complements courses in Object-Oriented Programming, Database Management, and Final Year Projects. It connects theoretical knowledge with practical project execution, strengthening students' overall technical competence.</p> <p>Demand in the Industry:</p> <p>There is a growing global demand for software professionals skilled in both development and project management. Expertise in Agile methodologies, DevOps practices and software quality assurance is highly valued by employers globally.</p> <p>Job Prospects:</p> <p>Completing this course prepares students for roles such as Software Developer, Quality Analyst, System Designer, Project Assistant, and Scrum Team Member. With experience, they can advance to positions like Project Manager, Solution Architect, or Process Consultant.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Provide a strong foundation in software engineering principles, software process models, development methodologies, and the role of Generative AI in modern software engineering.</p> <p>CO(A) 2. Introduce techniques for software requirements engineering, system analysis, and UML-based modeling, including AI-assisted modeling approaches.</p> <p>CO(A) 3. Explain the fundamental concepts of software design principles, methods for accurately estimating project effort and cost, and scheduling techniques used to plan, track, and manage software development activities efficiently.</p>	

	<p>CO(A) 4. Introduce students to software quality assurance, risk management, and configuration management practices.</p> <p>CO(A) 5. Develop an understanding of software testing concepts and effective test case design techniques.</p>
8	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Describe various software process models and apply appropriate models, including AI-assisted approaches, to different project scenarios.</p> <p>CO2: Analyze software requirements and develop system models using UML.</p> <p>CO3: Implement software design principles and use estimation techniques, including AI-supported methods, in the development process.</p> <p>CO4: Effectively plan, schedule, and manage software projects following industry-standard and responsible AI practices.</p> <p>CO5: Demonstrate knowledge of quality assurance and conduct software testing using suitable methodologies.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Software Engineering: Understanding the nature and characteristics of software, fundamental principles and best practices of Software Engineering, Overview of Software Process Framework and Layered Technology. Role of Generative AI in Software Engineering, Ethical and Responsible Use of AI in Software Development. Introduction to Process Patterns, Basics of Capability Maturity Model Integration (CMMI). • Software Development Models: Exploration of prescriptive models such as Waterfall, Incremental, and Rapid Application Development (RAD), evolutionary models such as Prototyping and Spiral, Agile methodologies with emphasis on the: Agile Manifesto, Extreme Programming (XP), Scrum Overview. Introduction to DevOps and Continuous Integration/Continuous Deployment (CI/CD). • Requirements Engineering and Analysis: Overview of Requirements Engineering Process; Elicitation Techniques including Interviews, Workshops, and Use Cases; Components and Key Characteristics of a Software Requirements Specification (SRS) and Validation of Requirements. • System and Object-Oriented Modeling (using UML): Use Case Diagram, Class Diagram, Sequence Diagram, Activity Diagram, State Chart Diagram, Component & Deployment Diagram. Introduction to AI-assisted UML Modeling, automated model generation and validation from software requirements, and its role in modern software engineering. <p>UNIT II:</p> <ul style="list-style-type: none"> • Software Design and Architecture: Design Principles- Coupling and Cohesion,

Comparison of Functional-Oriented and Object-Oriented design approaches, System Architecture design concepts, Design Verification and Validation processes.

- **Software Metrics and Estimation:** Techniques for Software measurement including Lines of Code (LOC), Function Point Analysis, and Use Case based estimation, Overview of Object-Oriented Metrics, Introduction to COCOMO II, Estimation practices in Agile Development, Decision making considerations for Make/Buy Decision.
- **Software Project Management:** Project planning covering scope, feasibility analysis, and resource estimation, Project Scheduling through Effort Estimation, Time-Line Charts, Gantt Charts, Risk Management: Identification, Projection, RMMM (Risk Mitigation, Monitoring and Management) Plan.
- **Software Quality Assurance (SQA):** Introduction to SQA and its importance in software development. Covers activities such as reviews, formal technical reviews (FTR), reliability assessment, and SQA plan.
- **Software Testing:** Verification & Validation, Introduction to testing principles, objectives, and test oracles. Levels of testing including unit, integration, system, and acceptance testing, as well as white-box and black-box techniques. Includes preparation of test plans and test case design methodologies.
- **Automated Testing:** Overview of automated testing tools and techniques.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q.No.	Structure of the Questions	Marks
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Documentation	10
2.	Presentation	10
		Total 20

Select any one sample project* and prepare the following deliverables:

1. Problem Statement and SRS
2. DFD & Structured Chart
3. Use Case Diagram
4. Class Diagram, Object Diagram
5. State-chart & Activity Diagrams
6. Sequence Diagram, Collaboration Diagram
7. Component & Deployment Diagrams
8. Using Function Point (FP) Method
9. Gantt Chart / PERT Chart
10. Test Cases for Unit, Integration, White Box & Black Box Testing

List of sample projects:

- a) Student Result Management System
- b) Library management system
- c) Inventory control system
- d) Accounting system
- e) Fast food billing system
- f) Bank loan system
- g) Blood bank system
- h) Railway reservation system
- i) Automatic teller machine
- j) Video library management system
- k) Hotel management system
- l) Hostel management system
- m) Share online trading
- n) Resource management system
- o) Court case management system

11

REFERENCES:

1. Software Engineering, A Practitioner's Approach, Roger S, Pressman, 2019
https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://mlsu.ac.in/econtents/16_EBOOK-7th_ed_software_engineering_a_practitioners_approach_by_roger_s_pressman.pdf&ved=2ahUKEwjgwPLn7d2QAxXHV2wGHWSKPQ4QFnoECEYQAAQ&usg=AOvVaw14KwvDeRZU6u8gI6d1TsFB
2. Software Engineering: principles and Practices, Deepak Jain, OXFORD University Press, 2008
3. Software Engineering, Ian Sommerville, Pearson Education, 2017
https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://dn790001.ca.archive.org/0/items/bme-vik-konyvek/Software%2520Engineering%2520-%2520Ian%2520Sommerville.pdf&ved=2ahUKEwiCg5Dx192QAxUzgK8BHfiEFvEQFnoECHIQAAQ&usg=AOvVaw1nGKn3K_GVDamqrvQlndyZ
4. Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI, 2018

- | | |
|--|---|
| | <ol style="list-style-type: none"><li data-bbox="331 197 1490 264">5. Software Engineering: Principles and Practices, Hans Van Vliet, John Wiley & Sons, 2010<li data-bbox="331 268 1308 302">6. A Concise Introduction to Software Engineering, Pankaj Jalote, Springer |
|--|---|

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Title: IoT Technologies

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: IoT Technologies
Course Code: CHMCSIV3

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Internet of Things (IoT) is a rapidly evolving domain that enables devices to connect, communicate, and automate tasks through data exchange. This course provides a foundational understanding of IoT, covering its architecture, evolution, core components, and the integration between the internet and physical devices.</p> <p>Relevance:</p> <p>As industries increasingly embrace automation and real-time analytics, understanding IoT has become essential for building intelligent and interconnected systems. Its applications extend across diverse sectors, including healthcare, agriculture, logistics, and smart city development.</p> <p>Usefulness:</p> <p>This course provides students with practical skills to build prototypes, interface sensors and actuators, and understand communication protocols. Through hands-on experience, learners develop the ability to design, implement, and troubleshoot IoT systems effectively.</p> <p>Application:</p> <p>Students apply their learning to real-world applications such as environmental monitoring, smart homes, precision farming, and intelligent transportation. The course builds foundational skills for designing and developing IoT-based solutions from the ground up.</p> <p>Interest:</p> <p>IoT presents a dynamic integration of hardware, software, and communication technologies, offering an engaging learning experience for students who enjoy practical problem-solving. Exploring microcontrollers and sensors provides a creative, hands-on approach to learning.</p>

		<p>Connection with Other Courses:</p> <p>This course complements subjects such as Embedded Systems, Computer Networks, Artificial Intelligence, Cloud Computing, and Mobile Application Development. It serves as a practical link between theoretical concepts and real-world applications.</p> <p>Demand in the Industry:</p> <p>The IoT industry is rapidly expanding, creating a strong demand for professionals skilled in embedded programming, cloud integration, sensor interfacing, and data security. Organizations seek experts capable of designing, deploying, and managing scalable IoT systems.</p> <p>Job Prospects:</p> <p>Upon completing this course, students can pursue roles such as IoT Developer, Embedded Systems Engineer, Firmware Developer, IoT Solution Architect, and Application Developer. With advanced specialization, opportunities extend to research and development, automation, and innovation labs across various global industries.</p>
2	Vertical 1	Major
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To introduce the fundamentals, evolution, architecture, and design principles of the Internet of Things (IoT).</p> <p>CO(A) 2. To examine various IoT platforms, sensors, actuators, and communication protocols.</p> <p>CO(A) 3. To provide an understanding of system-on-chip (SoC) architecture and interfacing methodologies.</p> <p>CO(A) 4. To develop the ability to design IoT systems and integrate them with web and cloud technologies.</p> <p>CO(A) 5. Enhance awareness of real-world IoT applications with AI-enabled</p>	

	<p>monitoring and prediction, along with emerging trends such as Edge and Fog Computing.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <ul style="list-style-type: none"> CO1: Understand the fundamental concepts, architecture, and design of IoT systems. CO2: Identify and utilize various sensors, actuators, and IoT development boards such as Raspberry Pi, Arduino, and NodeMCU. CO3: Apply suitable communication protocols to ensure reliable and secure data exchange. CO4: Design and implement basic IoT applications for data collection, processing, and visualization. CO5: Understand IoT applications in healthcare, agriculture, transportation, and smart cities with AI-enabled monitoring and prediction, along with current trends in Edge, Fog, and Cloud computing.
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • IoT Basics: Definition, Characteristics, Scope, History & Evolution, IoT vs M2M • IoT Architectures: Three-layer and Five-layer architectures, Physical vs Logical Design • Enabling Technologies: Cloud Computing, Big Data, AI, Embedded Systems – overview and role in IoT • IoT Components & Frameworks: Smart “Things” and identifiers, Overview of IoT Frameworks (AWS IoT, Google Cloud IoT, Azure IoT Hub) • System on Chip (SoC): Definition, Structure & Characteristics, Components (FPGA, GPU, APU, Compute Units), ARM & ATmega328 overview • IoT Hardware Platforms: Raspberry Pi, Arduino, NodeMCU – capabilities, application examples, platform selection criteria • Hardware Interfacing & Communication Protocols: Basic components (LED, Button, Motor, 8x8 LED Grid), Communication protocols: GPIO, PWM, UART, I2C, SPI <p>UNIT II:</p> <ul style="list-style-type: none"> • Sensors & Actuators: Digital & Analog Sensors: Temperature, Humidity, Motion, Light, Gas, Ultrasonic; Actuators: Relay Switch, Servo Motor – examples in real life • IoT Communication Protocols: HTTP, MQTT, CoAP, XMPP, UPnP – purpose, advantages, and real-world use cases.

	<ul style="list-style-type: none"> • IoT Security & Privacy: IoT security challenges, basic privacy concerns, examples of security threats • Web & Cloud Integration in IoT: -Web server setup, IoT device data exchange, Node-RED basics, Introduction to cloud platforms – AWS, Azure • Wireless Sensor Networks (WSNs): Basics, architecture, types, role in IoT communication, real-world scenarios • Edge & Fog Computing: Definition, Purpose and comparison with cloud, edge architectures, real-world examples • IoT Applications: Case Studies – Healthcare (patient monitoring and AI-based health prediction), Agriculture (smart irrigation with AI-enabled crop monitoring), Transportation (smart parking with AI-based traffic prediction and facial recognition), Smart Cities (traffic & energy management with AI-assisted smart surveillance).
--	---

10	<p>Scheme of Examination and Assessment Pattern Paper – 50 Marks External Examination: Semester End External - 30 marks Time: 01:00 hours Format of Question Paper</p> <table border="1" style="margin: auto;"> <thead> <tr> <th style="text-align: center;">Q.No.</th> <th style="text-align: center;">Structure of the Questions</th> <th style="text-align: center;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1.</td> <td>Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)</td> <td style="text-align: center;">15</td> </tr> <tr> <td style="text-align: center;">2.</td> <td>Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)</td> <td style="text-align: center;">15</td> </tr> <tr> <td colspan="2" style="text-align: center;">Total</td> <td style="text-align: center;">30</td> </tr> </tbody> </table> <p>Internal Examination: Continuous Evaluation – 20 marks</p> <table border="1" style="margin: auto;"> <thead> <tr> <th></th> <th style="text-align: center;">Assessment / Evaluation</th> <th style="text-align: center;">Marks</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1.</td> <td>Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes.</td> <td style="text-align: center;">10</td> </tr> <tr> <td style="text-align: center;">2.</td> <td>Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.</td> <td style="text-align: center;">10</td> </tr> <tr> <td colspan="2" style="text-align: right;">Total</td> <td style="text-align: center;">20</td> </tr> </tbody> </table>	Q.No.	Structure of the Questions	Marks	1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15	2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15	Total		30		Assessment / Evaluation	Marks	1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10	2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10	Total		20
Q.No.	Structure of the Questions	Marks																							
1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15																							
2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15																							
Total		30																							
	Assessment / Evaluation	Marks																							
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10																							
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10																							
Total		20																							

11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Introduction to IoT Paperback by Sudip Misra , Anandarup Mukherjee , Arijit Roy , Cambridge Press, 2022
-----------	--

- | | |
|--|---|
| | <ol style="list-style-type: none">2. Jain, Prof. Satish, Singh, Shashi, “Internet of Things and its Applications”, 1st Edition, BPB, 2020.3. Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, Internet of Things, Wiley, India, 20194. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 20205. Internet of Things by Vinayak Shinde, SYBGEN Learning India Pvt. Ltd, 20206. Internet of things, Dr. Kamlesh Lakhwani, Dr. Hemant kumar Gianey, Josef Kofi Wireko, Kamalkant Hiran, BPB Publication, 20207. Arduino, Raspberry Pi, NodeMCU Simple projects in easy way by Anbazhagan k and Ambika Parameswari k, 2019.8. Mastering the Raspberry Pi, Warren Gay, Apress, 20149. https://www.cavliwireless.com/blog/nerdiest-of-things/iot-smart-city-solutions10. chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://ijiird.com/wp-content/uploads/080212.pdf11. https://www.sciencedirect.com/science/article/pii/S2665972725000285 |
|--|---|

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Title: Computer Science Practical 4

**Vertical - 1
Major Course - 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Computer Science Practical 4

Course Code: CHMCSIV4

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>This major practical course integrates two rapidly evolving areas in computing — Computer Networking (CN) and the Internet of Things (IoT). The first part provides hands-on experience in network setup, routing, and traffic analysis, building a strong foundation in connectivity and data communication. The second part introduces students to intelligent connected devices, where they design and deploy smart systems using Raspberry Pi and Arduino, interface sensors and actuators, and implement cloud-based interactions for real-world applications.</p> <p>Relevance:</p> <p>This course holds great importance in today’s digital era, where interconnected devices and robust communication networks serve as the backbone of modern industries. By integrating Computer Networks and IoT, it equips students with comprehensive knowledge of data communication and real-time embedded systems. As automation and smart technologies continue to expand across sectors, the demand for professionals skilled in both areas is steadily rising.</p> <p>Usefulness:</p> <p>This course provides students with practical expertise in network configuration, hardware interfacing, and communication protocols for control and data exchange. It bridges theory and practice by enabling learners to design, simulate, and test complete systems. Through hands-on experience, students strengthen their understanding and develop a solid foundation for advanced studies or professional careers.</p> <p>Application:</p> <p>The knowledge acquired in this course can be directly applied to developing IoT solutions such as smart homes, environmental monitoring systems, healthcare devices, and automated machinery. Students also gain experience in simulating network infrastructures and implementing routing protocols, vital for building enterprise-level networks. The applications of this course extend across diverse domains, including agriculture,</p>

		<p>transportation, education, and urban development.</p> <p>Interest:</p> <p>This course sparks student interest through hands-on activities such as blinking LEDs, controlling motors, and streaming sensor data. Using tools like Cisco Packet Tracer, Node-RED, and Wireshark, learners gain interactive visual and analytical insights into both networking and IoT systems. The emphasis on experimentation and innovation ensures a dynamic and engaging learning experience</p> <p>Connection with Other Courses:</p> <p>This course complements core subjects such as Computer Networks, Microprocessors and Embedded Systems, Operating Systems, and Web Programming. It enables students to apply theoretical concepts in practical scenarios, fostering interdisciplinary learning. Additionally, its focus on programming and cloud communication connects closely with courses like Data Science and Cloud Computing.</p> <p>Demand in the Industry:</p> <p>There is an increasing demand for professionals proficient in IoT and network technologies across sectors such as healthcare, smart infrastructure, manufacturing, and IT services. Organizations seek individuals capable of developing, deploying, and maintaining systems that integrate hardware with software networking components. This course equips students with the skills that align with current industry needs and emerging technological trends.</p> <p>Job Prospects:</p> <p>Upon completing this course, students are prepared for roles such as IoT Developer, Network Administrator, Embedded Systems Engineer, and Cloud-IoT Integrator. They can pursue careers in fields like telecommunications, automation, research and development, and smart technologies. The practical experience gained enables them to perform effectively in both technical and interdisciplinary teams.</p>
2	Vertical 1	Major
3	Type	Practical

4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.Develop practical skills in computer networking tools, commands, and simulations using Cisco Packet Tracer.</p> <p>CO(A) 2.Examine various network topologies, IP addressing methods (static and dynamic), routing protocols (RIP, OSPF, and BGP), and VLAN-based network segmentation and communication.</p> <p>CO(A) 3.Apply IoT communication protocols, platforms, and hardware interfaces using Raspberry Pi, ESP32, MQTT, HTTP APIs, and Node-RED</p> <p>CO(A) 4.Implement and manage GPIO devices, sensors, cameras, LCD modules, and edge devices for real-world IoT monitoring and automation applications.</p> <p>CO(A) 5.Combine hardware and software tools such as Python, Node-RED, MQTT, HTTP services, and web dashboards to develop complete IoT solutions.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Execute and interpret network diagnostic and configuration commands in Windows and Linux environments for troubleshooting and connectivity analysis.</p> <p>CO2: Design and simulate wired and wireless networks in Cisco Packet Tracer using static/dynamic IP addressing, routing protocols (RIP, OSPF, BGP), and VLAN configurations.</p> <p>CO3: Analyze network traffic using Wireshark to examine protocol behavior, packet exchanges, and communication across different network layers.</p> <p>CO4: Set up and evaluate IoT hardware platforms using Raspberry Pi and ESP32 for GPIO interfacing, sensor integration, telemetry logging, and real-time monitoring applications.</p> <p>CO5: Build integrated IoT solutions using Node-RED, MQTT, HTTP APIs, web dashboards, and device communication for automation and monitoring applications.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p style="text-align: center;">Practical 1: Exploring Networking Commands via Windows CMD / LINUX Terminal</p> <p style="text-align: center;">a) Execute and observe the output of: ping, traceroute / tracert, netstat, arp, ipconfig / ifconfig, getmac, hostname, nslookup, pathping, systeminfo</p> <p style="text-align: center;">b) Discuss the purpose and interpretation of each command's output</p> <p style="text-align: center;">Learning Focus: Understanding diagnostic and configuration commands.</p>	

Practical 2: Building a Basic Peer-to-Peer Network

- a) Create a network with two PCs connected using a crossover cable in Cisco Packet Tracer.
- Assign static IP addresses
 - Test connectivity using ping and Packet Tracer simulation

Practical 3: Static IP Setup with One Server and Two Clients

- a) Connect 1 server and 2 computers using a switch (Cisco Packet Tracer).
- Use static IP addresses
 - Configure server services (e.g., HTTP or FTP) and test connectivity from clients.

Practical 4: Dynamic IP Allocation with Server and Clients

- a) Use DHCP service from a server to assign IPs to two PCs (Cisco Packet Tracer).
- Enable and configure DHCP on the server.
 - Verify IP allocation and connectivity using ipconfig.

Practical 5: Creating a Mixed Network with Wired and Wireless Devices

- a) One server, two wired PCs, and two mobile/wireless devices (Cisco Packet Tracer)
- Use appropriate cabling and access points
 - Assign IPs and test cross-device communication

Practical 6: RIP Version 1 Routing and RIP Version 2 Routing

- a) RIP Version 1 Routing Across Three Routers
- Three routers, each connected to at least three PCs (Cisco Packet Tracer)
 - Implement RIPv1 and RIPv2 routing between routers
 - Verify inter-network connectivity using ping and route tables
- b) RIP Version 2 Implementation
- Three routers, each connected to at least three PCs (Cisco Packet Tracer)
 - Enable RIPv2 and observe subnet mask handling
 - Use Packet Tracer's simulation mode to observe routing updates

Practical 7: FTP Client-Server Communication

- a) Two PCs and one server using FTP (Cisco Packet Tracer)

- Configure server IP and enable FTP service
- Assign client IP and connect to server via switch
- Test connectivity using ping
- Access FTP from client, list available files, and download files to client

Practical 8: BGP for Inter-domain Routing

- a) Three autonomous systems (AS) with routers running BGP
Three routers, each connected to at least three PCs (Cisco Packet Tracer)
- Configure BGP with different AS numbers
 - Establish peerings and test inter-AS communication

Practical 9: VLAN Configuration and Network Segmentation

- a) **One switch with four PCs configured in multiple VLANs**
- Create two VLANs, assign switch ports and IP addresses to PCs, verify communication within the same VLAN and
 - Verify restricted communication between different VLANs using ping

Practical 10: Protocol Analysis with Wireshark

- a) Set up network transactions for each protocol:
- ICMP (ping), TCP (web browsing), HTTP (via browser), UDP (DNS), FTP (file transfer)
- b) Apply filters and observe packet contents

UNIT II:

Practical 1: GPIO: LED Grid Module – Display Messages

- a) Write a Python program to accept a message from the user and scroll, change speed/brightness, display simple animations it across an 8x8 LED grid module.

Practical 2: LED GPIO fundamentals – patterns & PWM

- a) Write a Python program to implement the different patterns of ON/OFF an LED without requiring a physical button press.
- Steady ON/OFF
 - Slow blink
 - Fast blink
 - Binary counting
 - Chase pattern
 - PWM fade for LED brightness

Practical 3: Node-RED: Control LED via Internet of Things

- a) Create a Node-RED dashboard to control an LED remotely, allowing the user to turn it ON or OFF using the nodes.

Practical 4: System Telemetry Logger with LCD display

- a) Write a Python program to read IP, CPU%, memory%, and CPU temperature; display on 16x2 LCD scrolling for long text and append to a log file with timestamp

Practical 5: Node-RED: Simple HTTP Server Integration

- a) Write JavaScript functions in Node-RED to fetch data from an external HTTP server/API.
 - Process the data and display it in both the debug window and the dashboard (e.g., using a Gauge node).

Practical 6: Smart Parking System Using MQTT

- a) Simulate a smart parking system where parking sensors publish slot status via MQTT, and a central system subscribes to display slot availability.

Practical 7: Motion detector using PiCamera

- a) Write a Python program to use frame-difference motion detection; when motion detected, save a snapshot and print “Motion detected” on LCD; otherwise show “No motion”.

Practical 8: Node-RED Based Cryptocurrency Price Dashboard

- a) Fetch live Bitcoin and Ethereum prices in INR from the CoinGecko API.
- b) Visualize the prices on a Node-RED dashboard using gauges and charts.
- c) Trigger alerts when prices cross specified thresholds.

Practical 9: Fingerprint sensor enrollment & verification

- a) Write a Python program to enroll fingerprints using R307/R305 fingerprint sensor, verify, show LCD messages, blink LED, log results (timestamp, result) to CSV

Practical 10: Plant Soil Moisture Monitor using ESP32

- a) Design and simulate an IoT-based Plant Soil Moisture Monitoring System using ESP32.
- b) The system should simulate soil moisture levels using an analog input (potentiometer) instead of an external sensor.
- c) The ESP32 must read the simulated moisture value, determine if the soil is dry or moist, and display the result on a web dashboard hosted by the ESP32.

- d) Additionally, an LED should turn ON when the soil is dry (alert condition) and OFF when the soil is moist.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Kurose, J.F. & Ross, K.W. (2021). Computer Networking: A Top-Down Approach (7th ed.). Pearson Education India.
2. Forouzan, B.A. (2017). Data Communications and Networking (5th ed.). McGraw Hill Education India.
3. <https://www.netacad.com/courses/networking-basics?courseLang=en-US>
4. Ramya, V., & Shanmuga Priya, K. (2019). Practical Internet of Things: Concepts, applications and security. Chennai: Wiley India Pvt. Ltd.
5. Bahga, A. & Madisetti, V. (2014). Internet of Things: A Hands-on Approach. Universities Press India.
6. <https://nodered.org/docs/tutorials/>
7. Practical Node-RED Programming, by Taiji Hagino, Packt Publishing [https://unidel.edu.ng/focelibrary/books/Practical%20Node-RED%20Programming%20Learn%20powerful%20visual%20programming%20techniques%20and%20best%20practices%20for%20the%20web%20and%20IoT%20by%20Taiji%20Hagino%20\(z-lib.org\).pdf](https://unidel.edu.ng/focelibrary/books/Practical%20Node-RED%20Programming%20Learn%20powerful%20visual%20programming%20techniques%20and%20best%20practices%20for%20the%20web%20and%20IoT%20by%20Taiji%20Hagino%20(z-lib.org).pdf)
8. <https://www.codemag.com/article/1607071/Introduction-to-IoT-Using-the-Raspberry-Pi>
9. <https://magazine.raspberrypi.com/>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 2

Mathematical Modelling for Computer Applications : 2 Credits
**Practicals based on Mathematical Modelling for
Computer Applications : 2 Credits**

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year

Semester- IV

**Title: Mathematical Modelling for Computer
Applications**

**Vertical – 2
Minor Course – 2 Credits**

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

Title: Mathematical Modelling for Computer Applications
(Course Code: CHMMATHIV4)

Sr. No.	Heading	Particulars
1	Description of the Course:	This course provides an application-focused understanding of how mathematical models help explain and predict behaviours in modern computer systems. Students learn foundational modelling concepts and explore growth, decay, and spread processes relevant to digital platforms, user dynamics, and information or malware propagation. The course also introduces graph-based network models, influence and diffusion behaviours, and basic system and queue modelling used to analyse server performance and network congestion. Emphasis is placed on intuitive reasoning, real-world scenarios, and developing a modelling mindset for analysing complex computing environments.
2	Vertical 2	Minor
3	Type & Teaching Method	Theory
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives: By the end of this course, learners will be able to: <ul style="list-style-type: none"> CO(A) 1. Explain the fundamental principles of mathematical modelling, including model types, assumptions, formulation, validation, and interpretation, in the context of computer-based systems. CO(A) 2. Analyse and interpret growth, decay, and spread/epidemic processes (e.g., user growth, viral content, malware propagation) using conceptual and graphical representations of exponential, logistic, and SIR-type models. CO(A) 3. Apply graph-based modelling concepts such as connectivity, centrality, shortest paths, and influence propagation to understand real-world networked systems, including communication and social networks. CO(A) 4. Evaluate influence, diffusion, and cascading behaviours in digital platforms and networked environments, identifying the factors governing information spread, opinion formation, and viral phenomena. 	

	<p>CO(A) 5. Model and interpret basic system and queue behaviours—including arrival, service, waiting, and congestion—to understand performance characteristics of servers, cloud services, and networked systems.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of the course, learners will be able to:</p> <ul style="list-style-type: none"> CO1: Describe key modelling concepts—model types, assumptions, formulation, and validation—and explain their role in analysing computer-based systems. CO2: Interpret growth, decay, and epidemic/spread behaviours using conceptual, graphical, and application-oriented models relevant to digital platforms and networks. CO3: Apply basic graph and network modelling ideas (nodes, edges, paths, connectivity, centrality) to understand communication networks and social influence patterns. CO4: Analyse diffusion, influence, and cascading processes that govern viral information, trending content, and behavioural spread in online systems. CO5: Examine simple system and queue models to interpret server performance, congestion patterns, and latency behaviour in computing environments.
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I: Foundations, Growth Models and Spread/Epidemic Models</p> <ul style="list-style-type: none"> • Introduction to Mathematical Modelling: Purpose and role of models in computing. Types of models: deterministic/probabilistic, discrete/continuous. Steps in modelling: assumptions, formulation, validation, interpretation. Case discussions: system performance modelling, traffic growth, user dynamics • Growth and Decay Models (Application-Oriented): Exponential and logistic growth (conceptual, graphical understanding), Stability and long-term behaviour. Real-world applications such as: User-growth modelling for a digital platform, Forecasting server traffic or demand spikes, Modelling viral content growth • Epidemic and Spread Models in Computing: SIR-type models: compartments, flows, parameters. Epidemic thresholds, peak behaviour (conceptual). Real-world applications such as: Information/malware spread on networks, Misinformation propagation on social media, Contagion behaviour in distributed systems <p>UNIT II: Network Models, Influence Models and System Behaviour Models</p> <ul style="list-style-type: none"> • Network Models (Graph-Based Modelling): Graph concepts: nodes, edges, weighted/unweighted, paths. Connectivity, shortest paths, centrality (conceptual). Real-world applications such as : Communication and data networks, Routing intuition, Network congestion behaviour, Social network influence modelling (spread of opinions, influence scores) • Influence and Diffusion Models: Threshold models, cascading behaviour. Influence maximization (conceptual, no algorithms). Diffusion of ideas/information in networks. Real-world Applications such as: Viral marketing, Social influence dynamics, Trend propagation online • System Behaviour and Queue Models: Concept of discrete-time system modelling. Intuitive introduction to queues: arrival, service, waiting. Behaviour of servers under increasing load. Real-

world applications such as: Simple server/queue behaviour simulation, Cloud service delay modelling, Network latency and buffering

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

Q. No.	Structure of the Questions	Marks
1	Attempt ANY THREE out of SIX. (Each question of 5 marks, based on unit 1)	15
2	Attempt ANY THREE out of SIX. (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Class Test- It should be conducted using any Learning Management System such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes.	10
2.	Mini-Project, Case Study, Seminar, Posters, Self-Learning Evaluation, Presentation, etc.	10
		Total 20

11

REFERENCES:

1. Edwards, D., & Hamson, M. (1990). *Guide to mathematical modelling*. Springer.
2. Kapur, J. N. (1988). *Mathematical modelling*. Wiley Eastern.
3. Kermack, W. O., & McKendrick, A. G. (1991). *Contributions to the mathematical theory of epidemics* (Vol. 1). Springer.
4. Newman, M. E. J. (2010). *Networks: An introduction*. Oxford University Press.
5. Kleinrock, L. (1975). *Queueing systems, volume I: Theory*. Wiley.
6. Barabási, A.-L. (2016). *Network science*. Cambridge University Press.
7. Easley, D., & Kleinberg, J. (2010). *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press.
8. Pastor-Satorras, R., Castellano, C., Van Mieghem, P., & Vespignani, A. (2015). *Epidemic processes in complex networks*. Oxford University Press.
9. Trivedi, K. S. (2002). *Probability and statistics with reliability, queuing, and computer science applications* (2nd ed.). Wiley.
10. Mitzenmacher, M., & Upfal, E. (2005). *Probability and computing: Randomized algorithms*

and probabilistic analysis. Cambridge University Press.

11. Taha, H. A. (2017). *Operations research: An introduction* (10th ed.). Pearson.
12. Singiresu, S. R. (2016). *Engineering optimization: Theory and practice* (5th ed.). Wiley India.
13. Rao, S. S. (2010). *Applied numerical methods for engineers and scientists*. PHI Learning.
14. Kanti Swaroop, P., & Mohan, M. (2014). *Operations research*. Sultan Chand & Sons.
15. Negi, L. R. (2014). *Mathematical models in operations research*. McGraw Hill Education.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year

Semester- IV

**Title: Practical's based on Mathematical Modelling
for Computer Applications**

**Vertical – 2
Minor Course – 2 Credits**

(offered by Mathematics)

**with effect from
Academic Year 2025-2026**

Title: Practicals based on Mathematical Modelling for Computer Applications
(Course Code: CHMMATHIV5)

Sr. No.	Heading	Particulars
1	Description of the Course:	This course provides an application-focused understanding of how mathematical models help explain and predict behaviours in modern computer systems. Students learn foundational modelling concepts and explore growth, decay, and spread processes relevant to digital platforms, user dynamics, and information or malware propagation. The course also introduces graph-based network models, influence and diffusion behaviours, and basic system and queue modelling used to analyse server performance and network congestion. Emphasis is placed on intuitive reasoning, real-world scenarios, and developing a modelling mindset for analysing complex computing environments.
2	Vertical 2	Minor
3	Type & Teaching Method	Practical
4	Credit	2 Credits
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<p>CO(A) 1. To enable students to understand and apply the fundamental steps of mathematical model building—including problem formulation, assumption identification, and graphical interpretation—through practical, real-world computing scenarios.</p> <p>CO(A) 2. To develop the ability to construct, compare, and analyse growth-based models (exponential, logistic, and combined growth–decay) for applications such as user growth prediction, server load behaviour, and viral content dynamics.</p> <p>CO(A) 3. To equip learners with hands-on skills in designing and interpreting epidemic-style SIR models for understanding information/malware propagation and evaluating intervention or control strategies in networked systems.</p> <p>CO(A) 4. To introduce students to graph-based network modelling and analysis, enabling them to compute fundamental network metrics, identify bottlenecks,</p>

	<p>model routing behaviour, and understand influence dynamics in social networks.</p> <p>CO(A) 5. To apply modelling techniques to system performance scenarios by simulating queue behaviours, analysing server load patterns, and interpreting performance outcomes for computing and communication systems.</p>
<p>8</p>	<p>Course Outcomes: After successful completion of the course, learners will be able to:</p> <p>CO1: Apply the complete model-building process—problem identification, assumption setting, formulation, and visualization—to real computing scenarios such as user-growth prediction.</p> <p>CO2: Construct and compare exponential, logistic, and combined growth–decay models, and interpret their behaviour through graphical and analytical techniques relevant to server load and viral content dynamics.</p> <p>CO3: Implement and analyse SIR-based epidemic models to study malware or information spread in networks, and evaluate the impact of intervention or control strategies.</p> <p>CO4: Build and analyse graph-based network models, compute essential graph metrics, and identify network bottlenecks and routing behaviours through hands-on network simulations.</p> <p>CO5: Develop and execute practical simulations of influence diffusion, recommendation behaviour, and queue-based system performance models to interpret real-world server, congestion, and latency patterns.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>PRACTICAL 1: Introduction to Model Building - User Growth Prediction</p> <ul style="list-style-type: none"> a) Understanding the Model Building Process: Problem Formulation and Visualization b) Applying Different Model Types: Solving Simple Growth Prediction Problems <p>PRACTICAL 2: Exponential vs. Logistic Growth - Server Load Modelling</p> <ul style="list-style-type: none"> a) Formulation and Graphical Solution of Exponential Growth Problems b) Solving Logistic Growth Problems Using Graphical and Analytical Methods <p>PRACTICAL 3: Viral Content Spread Simulation</p> <ul style="list-style-type: none"> a) Implementation of Combined Growth-Decay Models for Viral Dynamics b) Application of Parameter Sensitivity Analysis to Viral Spread Models <p>PRACTICAL 4: SIR Epidemic Model - Malware Propagation</p> <ul style="list-style-type: none"> a) Understanding and Solving SIR Compartmental Models for Network Security b) Application of Control Measures and Intervention Strategies <p>PRACTICAL 5: Network Fundamentals - Social Network Analysis</p> <ul style="list-style-type: none"> a) Formulation of Network Models and Computing Basic Graph Metrics b) Application of Centrality Measures <p>PRACTICAL 6: Network Flow and Bottlenecks - Routing Model</p> <ul style="list-style-type: none"> a) Formulation and Solution of Maximum Flow Problems b) Comparison of Routing Strategies <p>PRACTICAL 7: Influence and Recommendation Systems</p> <ul style="list-style-type: none"> a) Implementation of Probabilistic Influence Spread Models in Social Networks

	<p>b) Application of Seed Selection Strategies and Adoption Rate Sensitivity Analysis</p> <p>PRACTICAL 8: System Queuing and Server Performance Modelling</p> <p>a) Mathematical Formulation and Manual Calculation of Queuing Systems</p> <p>b) Multi-Server Systems and Priority Queuing</p>																																																
<p>10</p>	<p style="text-align: center;">Scheme of Examination and Assessment Pattern Paper – 50</p> <p style="text-align: center;">External Examination: Semester End External - 30 marks</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Sr. No.</th> <th>Nature of Evaluation</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="text-align: center;">Project Evaluation:</td> </tr> <tr> <td style="text-align: center;">1)</td> <td>Problem definition and justification</td> <td style="text-align: center;">05</td> </tr> <tr> <td style="text-align: center;">2)</td> <td>Model design and assumptions</td> <td style="text-align: center;">05</td> </tr> <tr> <td style="text-align: center;">3)</td> <td>Implementation quality</td> <td style="text-align: center;">05</td> </tr> <tr> <td style="text-align: center;">4)</td> <td>Analysis depth</td> <td style="text-align: center;">05</td> </tr> <tr> <td style="text-align: center;">5)</td> <td>Presentation and documentation</td> <td style="text-align: center;">10</td> </tr> <tr> <td colspan="2"></td> <td style="text-align: right;">Total 30</td> </tr> </tbody> </table> <p>Internal Examination: Continuous Evaluation - 20 marks</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Sr. No.</th> <th>Nature of Evaluation</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td colspan="3" style="text-align: center;">Project Evaluation:</td> </tr> <tr> <td style="text-align: center;">1)</td> <td>Week 1: Choose topic, define problem, get approval</td> <td style="text-align: center;">03</td> </tr> <tr> <td style="text-align: center;">2)</td> <td>Week 2: Design model, start implementation</td> <td style="text-align: center;">05</td> </tr> <tr> <td style="text-align: center;">3)</td> <td>Week 3: Complete implementation, run scenarios</td> <td style="text-align: center;">06</td> </tr> <tr> <td style="text-align: center;">4)</td> <td>Week 4: Analysis, visualization (Interpretation and Validation)</td> <td style="text-align: center;">06</td> </tr> <tr> <td style="text-align: center;">5)</td> <td>Week 5: Report writing, Presentation and submission</td> <td style="text-align: center;">--</td> </tr> <tr> <td colspan="2"></td> <td style="text-align: right;">Total 20</td> </tr> </tbody> </table>	Sr. No.	Nature of Evaluation	Marks	Project Evaluation:			1)	Problem definition and justification	05	2)	Model design and assumptions	05	3)	Implementation quality	05	4)	Analysis depth	05	5)	Presentation and documentation	10			Total 30	Sr. No.	Nature of Evaluation	Marks	Project Evaluation:			1)	Week 1: Choose topic, define problem, get approval	03	2)	Week 2: Design model, start implementation	05	3)	Week 3: Complete implementation, run scenarios	06	4)	Week 4: Analysis, visualization (Interpretation and Validation)	06	5)	Week 5: Report writing, Presentation and submission	--			Total 20
Sr. No.	Nature of Evaluation	Marks																																															
Project Evaluation:																																																	
1)	Problem definition and justification	05																																															
2)	Model design and assumptions	05																																															
3)	Implementation quality	05																																															
4)	Analysis depth	05																																															
5)	Presentation and documentation	10																																															
		Total 30																																															
Sr. No.	Nature of Evaluation	Marks																																															
Project Evaluation:																																																	
1)	Week 1: Choose topic, define problem, get approval	03																																															
2)	Week 2: Design model, start implementation	05																																															
3)	Week 3: Complete implementation, run scenarios	06																																															
4)	Week 4: Analysis, visualization (Interpretation and Validation)	06																																															
5)	Week 5: Report writing, Presentation and submission	--																																															
		Total 20																																															
	<p>Integrated Project - Network Modelling Case Study: Apply all learned concepts to build a complete modelling project.</p> <p>Project Options (Choose ONE):</p> <ol style="list-style-type: none"> 1. Misinformation Spread on Social Media <ul style="list-style-type: none"> • Model how false information spreads through a social network. • Combine: SIR model + Network structure • Components: Believers (I), Skeptics (S), Fact-checkers (R) • Include: Network centrality, threshold effects, intervention strategies 2. User Growth and Churn on a Platform <ul style="list-style-type: none"> • Model user acquisition and retention for an app. • Combine: Logistic growth + Network effects • Components: New users, Active users, Churned users, Referrals • Include: Growth rate, retention rate, network-driven virality 3. Network Congestion and Load Balancing <ul style="list-style-type: none"> • Model traffic distribution in a server network. • Combine: Flow models + Queuing concepts • Components: Multiple servers, varying capacities, request routing 																																																

- Include: Bottlenecks, load distribution, optimization

4. Open Choice

- Propose your own computing-related modelling problem.
- Must incorporate at least TWO concepts from the course
- Must have clear application to computer science
- Get instructor approval before starting

Project Requirements for internals:

1. Problem Definition:

- Clear statement of problem
- Why is modelling useful here?
- What are you trying to predict/understand?

2. Model Design:

- Clearly state all assumptions
- Define model type and justify choice
- Explain parameters and their meanings
- Draw diagrams of model structure

3. Implementation:

- Build working simulation (Excel/Python)
- Show calculations/code clearly
- Demonstrate model runs with different parameters

4. Analysis:

- Present results with visualizations (graphs, charts)
- Compare different scenarios
- Identify key insights and patterns
- Discuss thresholds, peaks, optimal points

5. Interpretation and Validation:

- Do results make sense?
- What are model limitations?
- How could the model be improved?
- Real-world applicability

Deliverables for external evaluation:

1. Written Report (1500-2000 words):

- All sections above
- Professional formatting
- References to course concepts

2. Visualizations:

- Minimum 3 graphs/charts
- Network diagrams where applicable
- Clear labels and legends

3. Presentation (10 minutes):

- Problem overview
- Model explanation
- Key results
- Insights and recommendations

4. Working Model:

	<ul style="list-style-type: none"> ○ Excel file or Python code ○ Commented and organized ○ Runnable by instructor
<p>11</p>	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Edwards, D., & Hamson, M. (1990). <i>Guide to mathematical modelling</i>. Springer. 2. Kapur, J. N. (1988). <i>Mathematical modelling</i>. Wiley Eastern. 3. Kermack, W. O., & McKendrick, A. G. (1991). <i>Contributions to the mathematical theory of epidemics</i> (Vol. 1). Springer. 4. Newman, M. E. J. (2010). <i>Networks: An introduction</i>. Oxford University Press. 5. Kleinrock, L. (1975). <i>Queueing systems, volume I: Theory</i>. Wiley. 6. Barabási, A.-L. (2016). <i>Network science</i>. Cambridge University Press. 7. Easley, D., & Kleinberg, J. (2010). <i>Networks, crowds, and markets: Reasoning about a highly connected world</i>. Cambridge University Press. 8. Pastor-Satorras, R., Castellano, C., Van Mieghem, P., & Vespignani, A. (2015). <i>Epidemic processes in complex networks</i>. Oxford University Press. 9. Trivedi, K. S. (2002). <i>Probability and statistics with reliability, queuing, and computer science applications</i> (2nd ed.). Wiley. 10. Mitzenmacher, M., & Upfal, E. (2005). <i>Probability and computing: Randomized algorithms and probabilistic analysis</i>. Cambridge University Press. 11. Taha, H. A. (2017). <i>Operations research: An introduction</i> (10th ed.). Pearson. 12. Singiresu, S. R. (2016). <i>Engineering optimization: Theory and practice</i> (5th ed.). Wiley India. 13. Rao, S. S. (2010). <i>Applied numerical methods for engineers and scientists</i>. PHI Learning. 14. Kanti Swaroop, P., & Mohan, M. (2014). <i>Operations research</i>. Sultan Chand & Sons. 15. Negi, L. R. (2014). <i>Mathematical models in operations research</i>. McGraw Hill Education.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 3

Open Elective Course (OE) : 2 Credits

**with effect from
Academic Year 2025-2026**



HSNC Board's

Smt. Chandibai Himathmal Mansukhani College

(Autonomous)

(Affiliated to the University of Mumbai)

University College Code: 217 | JD Office: T14



Estd. Year
1965

Faculty of Interdisciplinary

List of Skill Based Open Electives for Second Year: Semester – IV

Sr. No.	Semester IV Subject
1	Digital Interface, Web Design And Publishing
2	3D Modeling And Character Animation Fundamentals
3	Advance Tools Of AI For Economics And Education - II
4	English For Leadership and Strategic Communication
5	Urbanization And Real Estate: Applied Urban Planning, Design And Sustainable Cities
6	Travel Agency And Tour Operators Business
7	Managing Family Wealth Through Family Office-IV
8	Advanced Web Designing & Portfolio Development
9	Basics Of Nutrition - 4
10	Reel Strategy And Influencer Management
11	Preforming Art- Dance-4
12	Data Analysis Project Based Approach
13	Strategic Political Communication, Digital Governance And AI-Driven Public Engagement Skills
14	Psychology Of Personal Relationship-II
15	Digital Society And Social Change
16	Mushroom Cultivation Training And Trading Level 4
17	Pranayama And Yogic Breathing Practices
18	Perfumery Course Level 4
19	Career Launchpad: Communication And Employability Skills
20	Beautician: Strategic Business Planning -IV
21	Current Trends In Fashion Designing: Financial Perspective Level 4
22	Basics Of Accounting-IV
23	Digital Marketing -IV
24	Online Trading For Investment Management



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 4

Skill Enhancement Course (SEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Title: MERN Stack Development

**Vertical - 4
Skill Enhancement Course (SEC): 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: MERN Stack Development
Course Code: CHMCSIV8

Sr. No.	Heading	Particulars
1.	Description of the course:	<p>Introduction</p> <p>The MERN Stack is a widely adopted JavaScript-based technology stack used for building modern full-stack web applications. It integrates MongoDB (NoSQL database), Express.js (backend web framework), React (frontend library for building user interfaces), and Node.js (JavaScript runtime environment). By enabling the use of JavaScript across the entire development pipeline—from client-side interfaces to server logic and database operations—the MERN stack offers a streamlined, efficient, and highly consistent development experience.</p> <p>Relevance</p> <p>In an era defined by cross-platform access, real-time interactions, and rapidly scalable digital products, MERN has emerged as one of the most industry-relevant full-stack frameworks. It is extensively used by startups, enterprises, SaaS platforms, and product-based companies. This course caters to the increasing demand for developers proficient in full-stack JavaScript development, ensuring learners stay aligned with contemporary tools, workflows, and industry practices.</p> <p>Usefulness</p> <p>The MERN Stack course equips learners to develop end-to-end, production-ready web applications using JavaScript across all layers. Students gain practical exposure to component-driven UI development, RESTful backend services, database connectivity, and state management. This unified approach simplifies the learning curve, enhances productivity, and helps learners become independent developers capable of handling both front-end and back-end tasks.</p>

Application

Students will be able to design and build responsive, interactive applications such as e-commerce systems, admin dashboards, social networking platforms, project management tools, online booking systems, and real-time chat applications. The course enables learners to implement essential functionalities, including routing, authentication, CRUD operations, API development, and cloud-ready deployment. The hands-on nature of the course encourages the creation of portfolio-grade projects demonstrating full-stack proficiency.

Interest

Hands-on labs, live coding demonstrations, and real-time application development ensure a highly engaging learning experience. Learners see immediate outputs, fostering curiosity, experimentation, and creative problem-solving. The modularity of the MERN stack allows learners to explore each component—database, backend services, front-end UI, and server logic—at their own depth or specialize based on interest.

Connection with Other Courses

This course builds on foundational subjects such as HTML, CSS, JavaScript, Web Programming, and Database Management Systems (DBMS). It also provides a strong base for advanced areas including Mobile App Development (using React Native), Cloud Deployment, Microservices, and DevOps pipelines. Students with prior exposure to object-oriented programming, REST APIs, and basic web design will find the transition smooth and intuitive.

Demand in the Industry

MERN Stack development is widely used in the software industry due to its speed, scalability, open-source ecosystem, and cost-effectiveness. Organizations increasingly seek developers who can work across both front-end and back-end layers, making MERN skills highly valuable. With JavaScript consistently ranking among the most in-demand

		<p>programming languages worldwide, MERN developers continue to enjoy strong and growing career opportunities.</p> <p>Job Prospects</p> <p>Learners can pursue a range of technical roles, including: MERN Stack Developer, Full Stack JavaScript Developer, React Frontend Developer, Node.js / Express Backend Developer, API Developer and UI/UX-oriented React Developer. Job opportunities exist in product companies, IT service firms, SaaS startups, digital agencies, and freelance/consulting markets. The skillset also empowers learners to independently build and deploy their own applications or digital products.</p>
2.	Vertical 4:	SEC
3.	Type:	Practical
4.	Credits:	2 credits (1 credit = 30 Hours of Practical work in a semester)
5.	Hours Allotted:	60 Hours
6.	Marks Allotted:	50 Marks
7.	<p>Course Objectives:</p> <p>CO(A) 1. To establish a solid understanding of the MERN (MongoDB, Express.js, React, Node.js) Stack as a complete full-stack web development framework.</p> <p>CO(A) 2. To enable learners to design and develop interactive, responsive, and data-driven web applications with effective integration between front-end and back-end components.</p> <p>CO(A) 3. To familiarize students with RESTful API creation, routing mechanisms, middleware usage, and database handling using modern JavaScript technologies.</p> <p>CO(A) 4. To encourage the development of real-time and highly interactive applications through component-based interfaces, asynchronous operations, and contemporary development workflows.</p> <p>CO(A) 5. To provide extensive practical exposure through hands-on labs, mini projects, and full-stack implementation tasks that strengthen real-world development skills.</p>	

8.	<p>Course Outcomes: After successful completion of this course, students would be able to –</p> <p>CO1: Create, develop, and deploy complete full-stack web applications using the MERN (MongoDB, Express.js, React, Node.js) technology stack.</p> <p>CO2: Design secure, efficient, and scalable back-end services and APIs using Node.js, Express.js, and associated middleware.</p> <p>CO3: Build interactive, responsive, and modular user interfaces using React and contemporary front-end development practices.</p> <p>CO4: Implement CRUD functionalities and data modelling techniques in MongoDB through the use of Mongoose.</p> <p>CO5: Seamlessly connect the front-end and back-end components to enable smooth data communication and deliver an enhanced user experience.</p>
9.	<p style="text-align: center;">Syllabus</p> <p>Unit-I</p> <p>Practical 1: Environment Setup and Node.js Basics</p> <ol style="list-style-type: none"> a) Install and configure Node.js, MongoDB, and development tools b) Understand Node.js fundamentals and npm c) Create basic Node.js applications using modules <p>Practical 2: Building REST APIs with Express.js</p> <ol style="list-style-type: none"> a) Create Express.js applications b) Implement RESTful routing c) Use middleware for request processing d) Build CRUD API endpoints <p>Practical 3: MongoDB Integration with Mongoose</p> <ol style="list-style-type: none"> a) Connect Node.js application to MongoDB b) Define Mongoose schemas and models c) Perform CRUD operations using Mongoose d) Implement data validation <p>Practical 4: Introduction to React and Components</p> <ol style="list-style-type: none"> a) Set up React development environment b) Create functional components c) Understand JSX syntax d) Implement state management with hooks e) Handle events and conditional rendering <p>Practical 5: React Forms and Component Communication</p> <ol style="list-style-type: none"> a) Build complex forms in React b) Implement form validation c) Manage parent-child component communication d) Handle multiple state variables

Unit-II:

Practical 6: React Router and Navigation

- a) Implement client-side routing
- b) Create multi-page React application
- c) Use dynamic routes and URL parameters
- d) Implement nested routes

Practical 7: API Integration with React

- a) Connect React frontend to backend APIs
- b) Fetch data using axios
- c) Handle loading and error states
- d) Display API data in components

Practical 8: Full MERN CRUD Application – Backend

- a) Build complete backend for a Student Management System
- b) Implement authentication basics
- c) Add advanced validation
- d) Create organized API structure

Practical 9: Full MERN CRUD Application – Frontend

- a) Build React frontend for Student Management System
- b) Create reusable components
- c) Implement form validation
- d) Connect to backend API

Practical 10: Deployment and Production Build

- a) Prepare application for production
- b) Deploy backend and frontend
- c) Configure environment variables
- d) Test deployed application

10.

Scheme of Examination and Assessment Pattern

Paper – 50

External Examination: Semester End External - 30 marks

Sr. No.	Nature of Evaluation	Marks
	Project Evaluation:	
1)	Functionality: All CRUD operations working	12
2)	Code Quality: Organization, best practices	06
3)	UI/UX: User-friendly, responsive design	05
4)	Deployment: Successfully deployed and accessible	03
5)	Documentation: Complete README and user guide	04
		Total 30

Internal Examination: Continuous Evaluation - 20 marks

Sr. No.	Nature of Evaluation	Marks
Project Evaluation:		
1)	Week 1: Choose topic, define problem, get approval	03
2)	Week 2: Design model, initial implementation	05
3)	Week 3: Implementation progress, CRUD scenarios	06
4)	Week 4: Analysis, visualization and Validation	06
5)	Week 5: Report writing, Presentation and submission	--
Total		20

Project Guidelines (MERN Stack Project)

Purpose:

The MERN Stack Project enables students to apply theoretical learning to a real-world software development project using MERN (MongoDB, Express.js, React.js, and Node.js). The project emphasizes practical implementation, problem-solving, teamwork, and professional development practices aligned with industry standards.

Learning Outcomes / Objectives:

- Apply interdisciplinary knowledge and technical concepts to design efficient solutions for real-world problems.
- Gain practical experience across all stages of the Software Development Life Cycle (SDLC): requirement analysis, design, development, testing, and deployment.
- Develop familiarity with global IT standards, documentation practices, and professional ethics.
- Enhance collaboration and project planning skills through proper task allocation and collaboration including communication.
- Prepare well-structured technical documentation according to academic and industry norms.
- Manage time, resources, and responsibilities effectively for successful project completion.

Types of Projects

Students may choose any one of the following project categories:

1. Real-world Solution Development:

Projects that solve an existing problem using a MERN-based application.

Examples:

- Online Banking Portal with secure authentication
- Timetable management interface
- Healthcare appointment booking and record management
- Cloud-based document management dashboard.
- Appointment scheduling system.

2. Innovative Product Development:

Projects introducing a new and unique concept with practical use or focuses on creating a new idea or product with unique features or utility.

Examples:

- Smart Energy Monitoring Dashboard using IoT data (visualized via MERN)
- Predictive Maintenance Web App for machines

3. **Research-oriented Projects:**

Projects exploring advanced or emerging technologies and integrating research findings into a MERN application.

Examples:

- Image recognition dashboard using MERN with an ML API
- Business forecasting, data visualization and analytics portal

Tools and Technologies

Students are encouraged to explore and use relevant tools and frameworks. Open-source technologies are preferred.

- **Frontend:** React.js
- **Backend:** Node.js + Express.js
- **Database:** MongoDB (Atlas or local)
- **Other tools:** Git / GitHub, VS Code, Postman, hosting services (Render, Netlify, Vercel, etc.)

Evaluation will focus on the project outcome and implementation quality—not the tools selected.

Project Supervision

Each group will be assigned a faculty guide who will:

- Approve the project proposal
- Provide technical assistance and feedback
- Review project progress and advise on improvements
- Ensure ethical and professional conduct throughout the project

Team Size

- **1 (Individual)**

Project Report Guidelines

A spiral bound copy of the project report must be submitted. The documentation should be formally certified by the project guide and HoD.

Report structure:

- Title Page (as per template)
- Certificate Page (signed by guide and HoD) — **Certified Project Documentation**
- Declaration (signed by student)
- Acknowledgement
- Table of Contents
- Chapters (as applicable):
 - Introduction and Problem Statement

	<ul style="list-style-type: none"> ○ System Study / Analysis ○ System Design (ERD, UML diagrams, architecture diagrams) ○ Implementation (code snippets/screenshots) ○ Testing (test cases, results) ○ Deployment (URL, hosting details, build steps) ● List of Figures / Tables (if applicable) ● References (IEEE / Springer format) ● Appendices (datasets etc.) <p>Formatting:</p> <ul style="list-style-type: none"> ● Font: Times New Roman ● Font Size: 12 pt (text), 20 pt bold all caps (chapter headings) ● Spacing: Single line spacing
11.	<p>REFERENCES:</p> <p>13. Tomasz Dyl , Kamil Przeorski , Maciej Czarnecki (2017). <i>Mastering full stack React web development</i>. Packt Publishing.</p> <p>14. Subramanian, V. (2019). <i>Pro MERN stack: Full stack web app development with Mongo, Express, React, and Node</i> (2nd ed.). Apress.</p> <p>Official Documentation:</p> <ul style="list-style-type: none"> ● Node.js: https://nodejs.org/docs ● Express.js: https://expressjs.com ● MongoDB: https://docs.mongodb.com ● Mongoose: https://mongoosejs.com/docs ● React: https://react.dev ● React Router: https://reactrouter.com <p>Tools:</p> <ul style="list-style-type: none"> ● VS Code: https://code.visualstudio.com ● Postman: https://www.postman.com ● MongoDB Compass: https://www.mongodb.com/products/compass ● Git: https://git-scm.com <p>Deployment Platforms:</p> <ul style="list-style-type: none"> ● Render: https://render.com ● Vercel: https://vercel.com ● Netlify: https://www.netlify.com ● MongoDB Atlas: https://www.mongodb.com/cloud/atlas

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 5

Ability Enhancement Course (AEC) : 2 Credits

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year B.Sc.

Semester- IV

Vertical – 5

**Ability Enhancement Course (English)
2 Credits**

**(To be offered to Students who
opted Sindhi AEC in Sem I & II)**

**with effect from
Academic Year 2025-2026**

Title: Advanced English for Workplace and Academic Communication
Course Code: CHMBSCAECIV

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>In an increasingly competitive academic and professional landscape, learners require advanced communication skills that enable clarity, precision, critical thinking, and professionalism. This course focuses on practical, real-world communication abilities needed for college-level academic work, job applications, workplace collaboration, and digital interactions.</p> <p>Through hands-on tasks, real-world assignments, and communication practice, learners become adept in using English confidently and appropriately in diverse settings.</p>
2	Vertical 5	AEC: Advanced English for Workplace and Academic Communication
3	Type Teaching Methods:	Theory+ Practicum (Lecture/ Discussion/ Presentation/ Reading sessions/ Worksheets/ etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A)1: To develop advanced communication skills required for academic and professional success.</p> <p>CO(A)2: To train learners in report writing, summary writing, and formal documentation.</p> <p>CO(A)3: To enhance proficiency in digital and virtual communication platforms.</p> <p>CO(A)4: To strengthen presentation, interview, and workplace communication skills.</p> <p>CO(A)5: To build confidence in expressing ideas clearly to varied audiences.</p>	
8	<p>Course Outcomes: After completing this course, learners will be able to:</p> <p>CO-1: Demonstrate clarity, precision, and professionalism in communication.</p> <p>CO-2: Interpret and summarize written texts, visuals, and data accurately.</p> <p>CO-3: Prepare well-structured reports, emails, and professional documents.</p>	

CO-4: Use digital tools and virtual communication etiquette effectively.
CO-5: Communicate confidently in interviews, presentations, and teamwork situations

9

Syllabus

UNIT I: Communication for Academic & Professional Settings (15 Hours)

A. Communication in Academic & Workplace Contexts

1. Features of formal communication
2. Audience-centered communication
3. Ethics in communication: integrity, attribution, clarity
4. Explaining concepts in simple and clear language
5. Interpreting graphs, charts, tables, and infographics
6. Summarizing data concisely

B. Grammar & Style for Professional Writing

1. Tone: formal, neutral, objective
2. Avoiding redundancy and ambiguity
3. Active vs. passive structures
4. Editing, revising, and proofreading techniques

UNIT II: Practical Documentation & Employability Skills (15 Hours)

1. Report writing (academic/field-based/observational)
2. Project summary reports
3. Preparing short presentations
4. Creating informational posters or digital slides
5. Writing a formal complaint or request email
6. Creating a short informational or awareness write-up

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

All questions are compulsory:

Question No	Nature of Questions	Marks
Q. 1	Short Notes (Attempt any 3 out of 5) - Unit 1 OR Attempt Essay Type question. (1 out of 2) - Unit 1	15
Q. 2	Short Notes (Attempt any 3 out of 5) - Unit 2 OR Attempt Essay Type question. (1 out of 2) - Unit 2	15
Total		30

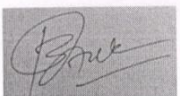


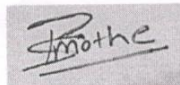
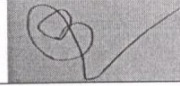
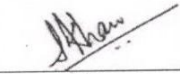
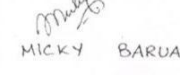

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Assignments on any one of the following topics: Academic/Scientific Report/ Informational Poster / Digital Infographic (Students are required to use AI assistance in the preparation of their drafts. Eg: Notion AI, Otter.ai, Grammarly, Google Gemini, Canva, Piktochart, etc)	15
2.	Class Attendance and Participation	05
	Total	20

11**References:**

1. Adler, Ronald B., et al. *Understanding Human Communication*. 15th ed., Oxford UP, 2021.
2. Bailey, Stephen. *Academic Writing: A Handbook for International Students*. 5th ed., Routledge, 2018.
3. Cargill, Margaret, and Patrick O'Connor. *Writing Scientific Research Articles*. Wiley-Blackwell, 2013.
4. Eastwood, John. *Oxford Guide to English Grammar*. Oxford UP, 2005.
5. Gerson, Sharon J., and Steven M. Gerson. *Technical Communication: Process and Product*. 9th ed., Pearson, 2021.
6. Hewings, Martin. *Advanced Grammar in Use*. Cambridge UP, 2013.
7. Kumar, Sanjay, and Pushp Lata. *Communication Skills*. 2nd ed., Oxford UP, 2018.
8. McCarthy, Michael, and Felicity O'Dell. *Academic Vocabulary in Use*. Cambridge UP, 2008.
9. Nordquist, Richard. *The Essentials of English Grammar*. McGraw-Hill Education, 2016.
10. Seely, John. *The Oxford Guide to Writing and Speaking*. Oxford UP, 2005.
11. Zinsser, William. *On Writing Well: The Classic Guide to Writing Nonfiction*. Harper Perennial, 2016.

Syllabus Committee:

Sr. No	Name of the Faculty	Designation and College	Signature
1.	Prof. (Dr.) Kailas Aute	Professor & Head, Dept. of English, Smt. CHM College	
2.	Prof. (Dr.) B. R. Hiramani,	(VC Nominee, University of Mumbai) Pancham Khemraj College, Sawantwadi	
3.	Prof. (Dr.) Vikas Raskar	(Subject Expert outside University) Hutatma Rajguru Mahavidyalay, Rajguru Nagar, Khed, (Affiliated to Savitribai Phule University)	
4.	Prof. (Dr.) Prashant Mothe	(Subject Expert outside University) Aadarsh Mahavidyalay, Umerga, Dharashiv, (Affiliated to Dr. Baba Saheb Ambedkar Marathwada University)	
5.	Mr. Ananda Pandhare	Asst. Professor, Dept. of English, Smt. CHM College	
6.	Ms. Sana Khan	Asst. Professor, Dept. of English, Smt. CHM College	
7.	Dr. Micky Barua	Faculty Vidyalkar Institute of technology, Alumni Member	 MICKY BARUA
8.	Ms. Sofy Verghese	Accenture, Industry Representative	

Name & Signature of the Ad-hoc BoS Chairperson: Prof. (Dr.) Kailas Aute



Name & Signature of the Dean: Prof. (Dr.) Nitin Arekar



**Smt. Chandibai Himathmal Mansukhani
College**

(Autonomous)

Second Year B.A

(Hindi)

Semester – IV

Title : हिंदी भाषा : व्यावहारिक प्रयोग

Vertical - 5

AEC – 2 Credits

with effect from

Academic Year 2025-2026

Title : हिंदी भाषा : व्यावहारिक प्रयोग**Course Code : CHMAECHINIV**

Sr.No.	Heading	Particulars
1.	Description of the Course :	<p>भाषा का जीवन में सदैव महत्व रहा है, जीवन और भाषा का चोली – दामन का संबंध है, जब हमारी भाषा मधुर और सार्थक होती है तो श्रेता पर विशिष्ट प्रभाव पड़ता है, भाषा का यदि सही और सार्थक रूप से प्रयोग किया जाए तो मुनष्य जीवन में कहीं भी असफल नहीं हो सकता है, इसी भाषा के माध्यम से हम सभी को अपनी ओर आकर्षित भी करते हैं, वर्तमान युग में रोजगार में बहुत से क्षेत्र भाषा से जुड़े हुए हैं, जिसके माध्यम से विद्यार्थी इनका लाभ ग्रहण कर सकते हैं, भाषाई क्षमता हमारे विचारों की संवाहक होती है, आज डिजिटल युग में अभिव्यक्ति के कई माध्यमों का प्रसार हुआ है, इन माध्यमों में भाषा ही सशक्त तत्व है जो आपकी अभिव्यक्ति को पूरे जगत को अवगत कराती है, भाषा का महत्व हर समय, हर माध्यम में रहा है, परंतु भाषा का सार्थक रूप का प्रयोग आज बहुत आवश्यक है। आज हिंदी अंतरराष्ट्रीय स्तर पर प्रयोग में लाई जा रही है, तकनीक, सूचना प्रौद्योगिकी सोशल मीडिया, राजनीति की भाषा हिंदी बन चुकी है, जीवन में कई क्षेत्रों में व्यावहारिक स्तर पर हमें अपनी भाषा के लिखित स्वरूप के कार्यों को करना होता है और ऐसे में कार्य-दक्षता महत्व रखती है, हिंदी भाषा में व्यावहारिक प्रयोग को केंद्र में रखकर और इन्हीं पहलुओं को ध्यान में रखते हुए इस पाठ्यक्रम का गठन किया गया है, हम हिंदी भाषा को सही और शुद्ध रूप में प्रयोग कर अभिव्यक्ति को सफल बनाए और बिना व्याकरण के यह संभव नहीं है, इस दृष्टि से पाठ्यक्रम सर्वाधिक लाभकारी सिद्ध होगा</p>
2.	Vertical : 5	AEC
3.	Type : Teaching Methods :	Theory + Practium Lecture / Discussion / Presentation / Self Study, etc.

4.	Credit :	2 Credits (1 Credit = 15 Hours for Theory)
5.	Hours Allotted :	30 Hours
6.	Marks Allotted :	50 Marks
7.	Course Objectives : CO(A)1: विद्यार्थियों को राजभाषा हिंदी का विधिवत ज्ञान प्रदान करना । CO(A)2: विद्यार्थियों को राजभाषा हिंदी के संवैधानिक महत्त्व से परिचित करवाना । CO(A)3: विद्यार्थियों को संज्ञा आदि का ज्ञान प्रदान करना । CO(A)4: विद्यार्थियों को कारकों, वाक्य रचना एवं भाषिक चिन्हों आदि का ज्ञान प्रदान करना ।	
	Course Outcomes : CO1 : विद्यार्थियों को राजभाषा हिंदी का ज्ञान प्राप्त होगा, एवं दक्षता प्राप्त होगी । CO2 : विद्यार्थियों को राजभाषा हिंदी के संवैधानिक महत्त्व की जानकारी प्राप्त होगी । CO3 : विद्यार्थियों को हिंदी – संज्ञा आदि का ज्ञान प्राप्त होने के साथ भाषा के शुद्ध, व्यावहारिक रूप का ज्ञान होगा । CO4 : विद्यार्थियों को कारकों, वाक्य रचना एवं भाषिक चिन्हों आदि का ज्ञान प्राप्त होगा ।	
9.	Syllabus	
	UNIT I :	
	1. हिंदी भाषा – सामान्य परिचय	
	2. राजभाषा हिंदी – संवैधानिक महत्त्व	
	3. वर्णमाला – स्वर एवं व्यंजन	
	4. शब्द भेद – सामान्य परिचय (संज्ञा आदि)	
	UNIT II :	
	1. वाक्य – सामान्य परिचय	
	2. वर्तनी – शुद्धता का प्रयोग एवं सावधानियाँ	
	3. कारक एवं विराम चिन्ह	
	4. निबंध लेखन – (सामाजिक निबंध, आत्मकथात्मक निबंध, समसामायिक निबंध)	

10.

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination : Semester End External – 30 Marks Time : 1:00 Hours

Format of Question Paper

All Questions are Compulsory

मूल्यांकन प्रारूप	इकाई	अंक
बाह्य मूल्यांकन		
प्रश्न 1 : चार प्रश्नों में से किन्हीं दो प्रश्नों के उत्तर लिखिए ।	इकाई 1	15
प्रश्न 2 : चार प्रश्नों में से किन्हीं दो प्रश्नों के उत्तर लिखिए ।	इकाई 2	15
	कुल अंक	30

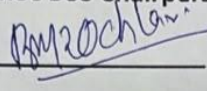
मूल्यांकन प्रारूप	अंक
आंतरिक मूल्यांकन	
<ul style="list-style-type: none"> ● AI की सहायता से हिंदी भाषा, राजभाषा हिंदी एवं उसके संवैधानिक महत्व पर शोध, सारांश एवं डिजिटल प्रस्तुति (Presentation) तैयार करना। ● AI Grammar एवं Writing Tools का उपयोग करके वर्णमाला, शब्द-भेद, वाक्य-रचना, कारक, विराम-चिह्न तथा वर्तनी शुद्धता का अभ्यास करना एवं त्रुटियों का विश्लेषण करना। ● AI की सहायता से हिंदी व्याकरण पर आधारित प्रश्नसंच (Question Bank), Quiz तथा Interactive अभ्यास-पत्र तैयार करना। ● AI Writing Tools का उपयोग करके सामाजिक, आत्मकथात्मक एवं समसामयिक विषयों पर निबंध लेखन करना तथा AI द्वारा तैयार निबंध और विद्यार्थी द्वारा लिखे गए निबंध का तुलनात्मक विश्लेषण करना। ● AI की सहायता से कठिन शब्दों के अर्थ, पर्यायवाची, विलोम, शब्दावली तथा सरल भाषा में व्याख्या तैयार करना। ● AI Voice Tools का उपयोग करके शुद्ध उच्चारण, वाचन एवं मौखिक अभिव्यक्ति का अभ्यास करना तथा उच्चारण संबंधी Feedback प्राप्त करना। 	20
कुल अंक	20

11.	<p>संदर्भ ग्रंथ सूची –</p> <ol style="list-style-type: none">1. बाबूराम सक्सेना – सामान्य भाषा विज्ञान, हिंदी साहित्य सम्मेलन, प्रयाग ।2. कामताप्रसाद गुरू – हिंदी व्याकरण, लोकभारती प्रकाशन, इलाहाबाद ।3. आचार्य देवेन्द्र नाथ शर्मा – भाषा विज्ञान की भूमिका, राधाकृष्ण प्रकाशन, दिल्ली ।4. भाषा विज्ञान एवं भाषा शास्त्र – कपिलदेव द्विवेदी, विश्वविद्यालय प्रकाशन, वाराणसी ।5. भोलानाथ तिवारी, भाषा विज्ञान, किताब महल, इलाहाबाद ।
-----	---

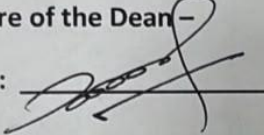
BoS in Hindi :

Sr No	Name of the Faculty	Designation and College
1.	Dr. Bhavna M.Rochlani	I/C HOD Asst. Professor CHM College Ulhasnagar
2.	Dr. Ajeet Kumar Rai	Associate Professor KC College Mumbai
3.	Dr. Santosh Motwani	Associate Professor RKT College Ulhasnagar

Name & Signature of the Ad-hoc BoS Chairperson -

Dr. Bhavna M. Rochlani : 

Name & Signature of the Dean -

Dr. Nitin Arekar : 



**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year BA/BCom/BSc/SFC
(Marathi)**

Semester- IV

**Vertical -5
Ability Enhancement Course (AEC) -2 Credits**

**with effect from
Academic Year 2026-2027**

Title: लेखन कौशल्ये - २ (महाजालावरील लेखन)

COURSE CODE: CHMAECMARIII

Sr. No.	Heading	Particulars
1	Description the Course:	<p>राष्ट्रीय शैक्षणिक धोरण - २०२० विद्यार्थ्यांच्या सर्वांगीण विकासावर (Wholistic Development) भर देते. या धोरणात सर्वांगीण विकासाचा भाग म्हणून क्षमता वर्धन अभ्यासक्रम (Ability Enhancement Course) या स्तंभांतर्गत भाषिक कौशल्य अभ्यासक्रमाचा समावेश करण्यात आला आहे. कला, वाणिज्य व विज्ञान या विद्याशाखांमध्ये अध्ययन करणाऱ्या विद्यार्थ्यांना तिसऱ्या सत्रामध्ये 'आधुनिक भारतीय भाषां'चे अध्ययन अनिवार्य करण्यात आले आहे. सदर क्षमता वर्धन अभ्यासक्रमाचे स्वरूप प्रामुख्याने भाषाकेंद्री असावे, असेही राष्ट्रीय शैक्षणिक धोरणात नमूद करण्यात आले आहे. विद्यार्थ्यांना विविध प्रकारच्या भाषिक कौशल्यांचा तपशीलवार परिचय करून देणे, तसेच ती कौशल्ये आत्मसात करण्याची संधी विद्यार्थ्यांना उपलब्ध करून देणे, ही या अभ्यासक्रमाची महत्त्वाची उद्दिष्टे आहेत. ही उद्दिष्टे लक्षात घेऊन 'लेखन कौशल्ये - २ (महाजालावरील लेखन)' (श्रेयांकने २) या अभ्यासपत्रिकेची आखणी करण्यात आली आहे.</p> <p>आंतरमहाजाल हे एकविसाव्या शतकातील अत्यंत प्रभावी साधन आहे. जगभरातील संगणक एकमेकांशी जोडले जाऊन त्यांचे जाळे तयार झाले आहे. विविध सामाजिक माध्यमस्थळांवर स्वतःचे खाते (अकाउंट) तयार करणे आणि त्यावर मराठी भाषा व देवनागरी लिपीतून लिहिणे, ही समकालीन संपर्क व्यवहारातील आवश्यक बाब झाली आहे. यास अनुसरून आपल्या अभिव्यक्तीला व्यासपीठ मिळवून देणारी अनुदिनी (ब्लॉग) तयार करणे, विकिपीडियावर भोवतालातील भाषा, साहित्य, संस्कृतीशी निगडित माहितीपर व विश्लेषणात्मक नोंदी लिहिणे, सामाजिक माध्यमस्थळांवरील आपल्या खात्यावर सातत्याने अभ्यासपूर्ण लेखन करणे, स्वक्षमतेशी निगडित समाजगट / आभासी कट्टे (कम्युनिटी ग्रुप) तयार करणे, या बाबींसाठी आवश्यक सामाजिक माध्यमस्थळ साक्षरता आणि मराठी भाषा व देवनागरी लिपीतून लिहिण्याची क्षमता 'लेखन कौशल्ये - २ (महाजालावरील लेखन)' (श्रेयांकने २) या अभ्यासपत्रिकेच्या अध्ययनातून विद्यार्थ्यांमध्ये निर्माण होईल.</p>
2	Vertical 5	Ability Enhancement Course
3	Type	Theory
4	Credit	2 Credits (1 Credit = 15 Hours for Theory or 30 Hours of Practical Work in a Semester)
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks

7	<p>Course Objectives:</p> <p>CO1: महाजालावरील लेखन कौशल्याचे स्वरूप समजावून सांगणे.</p> <p>CO2: महाजालावर प्रभावी लेखन करण्यासाठी आवश्यक असणाऱ्या तंत्रांचा परिचय करून देणे.</p> <p>CO3: नेहमीच्या पठडीतील लेखन व महाजालावरील लेखन यांमधील साम्य-भेद स्पष्ट करणे.</p> <p>CO4: विविध सामाजिक माध्यमस्थळांवर लेखन करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित करणे.</p>
8	<p>Learning Outcomes:</p> <p>प्रस्तुत अभ्यासक्रम शिकल्यानंतर:</p> <p>LO1: विद्यार्थ्यांना महाजालावरील लेखन कौशल्याचे स्वरूप समजेल.</p> <p>LO2: विद्यार्थ्यांना महाजालावर प्रभावी लेखन करण्यासाठी आवश्यक तंत्रांचा परिचय होईल.</p> <p>LO3: विद्यार्थ्यांना नेहमीच्या पठडीतील लेखन व महाजालावरील लेखन यांमधील साम्य-भेद स्पष्ट होईल.</p> <p>LO4: विद्यार्थ्यांमध्ये विविध सामाजिक माध्यमस्थळांवर लेखन करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित होतील.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I: सामाजिक माध्यमस्थळांवर मराठी भाषा व देवनागरी लिपीचा वापर करून लेखन (भाग - १)</p> <p>१. माध्यम साक्षरता</p> <p>२. अनुदिनी (ब्लॉग) लेखन</p> <p>३. विकिपीडियावरील लेखन</p> <p>(६० मिनिटांच्या १५ तासिका, श्रेयांकन १)</p> <p>(सूचना : विद्यार्थ्यांमध्ये उपरोक्त सामाजिक माध्यमस्थळांवर लेखन करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित होतील या दृष्टीने शिक्षकांनी सराव करून घ्यावा.)</p> <p>UNIT II: सामाजिक माध्यमस्थळांवर मराठी भाषा व देवनागरी लिपीचा वापर करून लेखन (भाग - २)</p> <p>१. फेसबुक, इन्स्टाग्राम, एक्स यांवरील लेखन</p> <p>२. समाज गट (कम्युनिटी ग्रुप), आभासी कट्टे यांवरील लेखन</p> <p>(६० मिनिटांच्या १५ तासिका, श्रेयांकन-१)</p> <p>(सूचना : विद्यार्थ्यांमध्ये उपरोक्त सामाजिक माध्यमस्थळांवर लेखन करण्यासाठी आवश्यक कौशल्ये व क्षमता विकसित होतील या दृष्टीने शिक्षकांनी सराव करून घ्यावा.)</p>

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00

hours

Format of Question Paper

All questions are compulsory:

Q. No	Nature of Questions	Marks
Q1	Essay type question on Module 1	10
Q2	Essay type question on Module 2	10
Q6	MCQs 15 out of 20, 10 MCQs on each module	10
		Total 30

Internal Examination: Continuous Evaluation - 20 marks

	Project and presentation / Viva	Marks
1.	<ul style="list-style-type: none"> विकिपीडियासाठी माहिती संकलन, संदर्भ व्यवस्थापन, तथ्य पडताळणी Fact Checking tool तयार करणे आणि लेख तयार करणे. AI साधने वापरून Facebook, Instagram आणि X (Twitter) साठी मराठीतील पोस्ट, Caption, Hashtags आणि Content Calendar तयार करणे. AI Copywriting Tools चा वापर करून सामाजिक माध्यमांसाठी आकर्षक व लक्षित (Targeted) मजकूर तयार करणे. AI साधनाच्या सहाय्याने Social Media Content चे भाषा विश्लेषण, शुद्धलेखन तपासणी आणि देवनागरी लिपीतील लेखन सुधारणा करणे. AI साधनाच्या सहाय्याने Community Groups आणि Virtual Discussion Platforms साठी माहितीपूर्ण पोस्ट, जनजागृती मोहीम आणि संवादात्मक मजकूर तयार करणे. AI Image Generation आणि Video Tools च्या सहाय्याने सामाजिक माध्यमांसाठी Creative Campaign तयार करणे. Blog, Wikipedia आणि Social Media Content यांची AI-assisted comparative analysis करून प्रभावी डिजिटल लेखनाचा अभ्यास करणे. <p>AI साधने: ChatGPT, Google Gemini, Claude, Perplexity AI, NotebookLM, Canva, CapCut, InVideo, Grammarly, QuillBot, Whisper, ElevenLabs</p>	20
		Total 20

11

संदर्भ ग्रंथ (Reference Books) :

१. मराठी व्याकरण आणि लेखन, विनायक गंधे व मीरा जोशी, निराली प्रकाशन, पुणे, २०१२.
२. उपयोजित मराठी, (संपा.) केतकी मोडक व अन्य, पद्मगंधा प्रकाशन, पुणे, २०१२.
३. मराठी भाषिक कौशल्य विकास, (संपा.) पृथ्वीराज तौर, अथर्व पब्लिकेशन्स, धुळे, २०१८.
४. व्यावहारिक मराठी, ल. रा. नसिराबादकर, भाषा विकास संशोधन संस्था, कोल्हापूर, २०२३.
५. *Aayushi International Interdisciplinary Research Journal* (ISSN 2349-638x) Peer Reviewed Journal www.aiirjournal.com

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Second Year B. Sc.
(Computer Science)**

Semester- IV

Vertical – 6

**Field Project (FP) : 2 Credits
Co-Curricular (CC) : 2 Credits**

**with effect from
Academic Year 2025-2026**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

Second Year

Semester- IV

Title: Field project

**Vertical - 6
Field Project 2 Credits**

**with effect from
Academic Year 2025-2026**

Title: Field Project

Sr. No.	Heading	Particulars
1	Description the Course:	The Field Project course, introduced under CHM Autonomy in alignment with the NEP 2020, aims to bridge theoretical knowledge with practical experience. It provides students with hands-on exposure to real-world socio-economic contexts through field visits, observation, and analysis in both urban and rural settings. By engaging directly with development-related issues, students enhance their research, problem-solving, and analytical skills while fostering social responsibility and environmental awareness. The course ultimately prepares learners for employability and active participation in nation-building.
2	Vertical 6	Field Project
3	Type & Teaching Methods	Field work
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	<ol style="list-style-type: none"> 1. To connect theoretical learning with real-world socio-economic contexts through practical field experiences. 2. To develop analytical, problem-solving, and teamwork skills in addressing contemporary social issues. 3. To cultivate an appreciation for research and its role in promoting societal and national development.
8	Learning Outcomes: students will be able to:	<p>LO1: Apply classroom knowledge to analyze real-life socio-economic challenges effectively.</p> <p>LO2: Demonstrate critical thinking, teamwork, and decision-making skills through field-based activities.</p> <p>LO3: Reflect on the relevance of research and experiential learning in contributing to social and national progress.</p>

Guidelines for Field Project

Following are the general guidelines for the conduct of Field Project (Semester III & IV)

Head of the Department (HOD)/ Field Project Co-ordinator

1. To ensure that FP program aligns with departmental and academic objectives as per NEP Structure within syllabus framework.
2. Appointment of field project incharges from the faculty of the department for group of Students.
3. To conduct orientation of FP Supervisor and decide the time line of the project.
4. To support the student for Filed Project.

FP Supervisor:

1. To give Guidelines for the field project.
2. To monitor student progress and provide guidance.

Project (Dissertation) Report:

Students are required to submit a report of the field project at the end of the semester in following suggested format.

The project should be typed on A4 sheets
 Font Size 12, Times New Roman, 1.5 line Spacing
 The project report shall have student details with signature of Field Project Incharge and photographs if any and it should be of minimum of 10 pages.

10

Scheme of Examination and Assessment Pattern

External Examination: Semester End External - 30 marks
Format of Question Paper

Nature of Evaluation	Marks
Field Project Report	30
Total 30	

Internal Examination: Continuous Evaluation - 20 marks

	Assessment / evaluation	Marks
1.	Involvement in Survey of Field Project /	05
2.	Field visit participation & completion	10
3.	Overall Impression	05
		Total 20

<p>11</p>	<p style="text-align: center;">Appendix I</p> <p style="text-align: center;">Attendance of the Student: Active Participation</p> <p>I, the undersigned Ms / Mr. _____ Roll No. ___ studying in the _____ Year of _____ Full-time Course is doing my project work under the guidance of Dr./Ms./Mr. _____, I wish to state that I have met my Internal guide on the following dates mentioned below for Project Guidance: -</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Sr.No.</th> <th style="text-align: center;">Date</th> <th style="text-align: center;">Signature of the Internal Guide</th> </tr> </thead> <tbody> <tr> <td style="height: 100px;"> </td> <td> </td> <td> </td> </tr> </tbody> </table> <p style="text-align: center;">_____</p> <p style="text-align: center;">Signature of the Candidate Supervisor</p> <p style="text-align: center;">_____</p> <p style="text-align: center;">Signature of Field Project Supervisor</p>	Sr.No.	Date	Signature of the Internal Guide			
Sr.No.	Date	Signature of the Internal Guide					
	<p style="text-align: center;">Appendix II</p> <p style="text-align: center;">Name of the Department/College/Institute</p> <p style="text-align: center;">Certificate</p> <p>I hereby certify that Mr./Ms. _____ Student of _____ studying in _____, has completed a project titled _____ in the area of _____ specialization for the academic year 2025-2026 to the best of my knowledge the work of the student is original and the information included in the project is correct.</p> <p style="text-align: center;">_____</p> <p style="text-align: center;">Field Project Supervisor</p> <p style="text-align: center;">_____</p> <p style="text-align: center;">Head of the Department/Principal</p>						

[Handwritten Signature]
Board of Examination

[Handwritten Signature]
**Principal & Chief Controller
Board of Examination**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

First Year

Semester- IV

Title: Cocurricular Course I

**Vertical - 6
Cocurricular Course - 2 Credits**

**with effect from
Academic Year 2025-2026**



HSNC Board's
Smt. Chandibai Himathmal Mansukhani College, Ulhasnagar
(Autonomous)
Affiliated to the University of Mumbai

Bachelor of Science
(Computer Science)
(Self-Financing Course)

Semester – V

Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2026-2027

Semester	Major		Minor	OE	VSC, SEC (VSEC)	AEC, VEC, IKS	OJT, FP, CEP, CC, RP	Cum. Cr. / Sem.
	Mandatory	Electives						
V	MJ11: Artificial Intelligence (TH) – 2	MJEL2: Wireless & Sensor Networks (TH) – 2 MJELP2: Wireless & Sensor Networks Practical (PR) – 2	4	-	VSC: 2 Ethical Hacking – 2	-	FP/CEP: 2	22
	MJ12: Cyber & Information Security (TH) – 2							
	MJP5: Computer Science Practical 5 (PR) – 2							
	MJP6: Mini Project – I (PR) – 2							
	MJ13: IKS in Computational Systems (TH) – 2							
	10	4						
VI	MJ14: Data Science (TH) – 2	MJEL3: Information Retrieval (TH) – 2 MJELP3: Information Retrieval Practical (PR) – 2	4	-	-	-	OJT:4	22
	MJ15: Cloud Computing (TH) – 2							
	MJ16: Software Project Management (TH) – 2							
	MJP7: Computer Science Practical 6 (PR) – 2							
	MJP8: Mini Project – II (PR) – 2							
	10	4						

[Abbreviation - OE – Open Electives, VSC – Vocation Skill Course, SEC – Skill Enhancement Course, (VSEC), AEC – Ability Enhancement Course, VEC – Value Education Course, IKS – Indian Knowledge System, OJT – on Job Training, FP – Field Project, CEP – Continuing Education Program, CC – Co-Curricular, RP – Research Project]

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Vertical – 1

Artificial Intelligence	: 2 Credits
Cyber and Information Security	: 2 Credits
IKS in Computational Systems	: 2 Credits
Computer Science Practical 5	: 2 Credits
Mini Project – I	: 2 Credits
Wireless & Sensor Networks	: 2 Credits
Wireless & Sensor Networks Practical	: 2 Credits

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Artificial Intelligence

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Artificial Intelligence

Course Code: CHMCSV1

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Artificial Intelligence course provides students with a foundation in the core principles that enable machines to exhibit intelligent behavior. It includes key topics such as intelligent agents, search strategies, logical reasoning, probabilistic models, machine learning approaches, reinforcement learning, and responsible AI practices. The course develops a systematic understanding of how intelligent systems are designed, evaluated, and implemented in real-world applications.</p> <p>Relevance:</p> <p>This course is highly relevant in today’s data-driven environment, where automation and intelligent decision-making play a crucial role across nearly all sectors. A strong understanding of AI helps bridge theoretical concepts in computer science with practical, real-world applications such as recommendation systems, predictive analytics, and autonomous technologies.</p> <p>Usefulness:</p> <p>The course provides students with practical and analytical skills needed to develop AI-based solutions. It strengthens problem-solving abilities through search algorithms, logical inference, probabilistic reasoning, and machine learning techniques. These competencies are vital for developers, analysts, researchers, and technology professionals working in the field of intelligent systems.</p> <p>Application:</p> <p>Artificial Intelligence concepts are extensively used in multiple fields such as healthcare diagnostics, financial prediction, robotics, natural language processing, cybersecurity threat identification, recommendation systems, and generative AI applications. AI also powers a wide range of modern technologies, including chatbots, fraud detection systems, autonomous vehicles, and smart virtual assistants, forming the core of today’s intelligent systems.</p> <p>Interest:</p>

		<p>The subject fosters curiosity by exploring how machines can learn, reason, and make decisions. It covers concepts such as A* search, game-playing algorithms like Minimax, Bayesian reasoning, neural networks, and reinforcement learning. These topics make the course intellectually engaging and closely aligned with emerging technologies, including deep learning and generative AI.</p> <p>Connection with Other Courses:</p> <p>AI concepts are closely connected with various core areas of computer science, including Data Structures and Algorithms (such as search trees and graphs), Discrete Mathematics (logic and probability), Data Science (model development and evaluation), Software Engineering (AI system lifecycle management), Cybersecurity (AI-based threat detection), and Cloud Computing (large-scale AI deployment). This course promotes a strong interdisciplinary understanding across the broader computer science curriculum.</p> <p>Demand in the Industry:</p> <p>The demand for Artificial Intelligence and Machine Learning skills is growing rapidly across industries such as IT services, fintech, healthcare, retail, manufacturing, and media. Organizations are seeking professionals with expertise in intelligent systems, predictive modeling, automated decision-making, and ethical AI implementation, making this course highly significant for industry preparedness.</p> <p>Job Prospects:</p> <p>Students skilled in Artificial Intelligence can pursue entry-level roles such as AI/ML intern, junior data analyst, AI application developer, or software engineer focused on integrating AI solutions. With higher qualifications and practical project experience, they can grow into advanced positions like machine learning engineer, data scientist, AI research associate, or AI product specialist.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)

4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To introduce the fundamental concepts of Artificial Intelligence along with the design and functioning of intelligent agents.</p> <p>CO(A) 2.To develop strong problem-solving abilities using search strategies and logical reasoning techniques.</p> <p>CO(A) 3.To build a clear understanding of probabilistic reasoning and machine learning methodologies.</p> <p>CO(A) 4.To introduce students to the concepts of reinforcement learning and latent variable models.</p> <p>CO(A) 5.To introduce Generative AI and Responsible AI, including prompt engineering, ethics, and evaluation using different prompting techniques.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Explain the core principles of intelligent agents, search strategies, and logical reasoning frameworks in Artificial Intelligence.</p> <p>CO2: Apply uninformed and informed search algorithms, along with adversarial reasoning techniques, to address and solve AI problems.</p> <p>CO3: Design, implement, and evaluate fundamental machine learning models for classification, regression, and clustering tasks.</p> <p>CO4: Analyze probabilistic models, including Bayesian networks and hidden variable models, to manage uncertainty in real-world applications.</p> <p>CO5: Evaluate AI systems using Responsible AI principles, covering Generative AI, prompt engineering, ethical issues, risks, and output assessment through different prompting techniques.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Foundations of AI & Intelligent Agents: What is AI? Rational agents vs human thinking, Computational agents, Agent–Environment interaction, Types of environments, Agent architectures (simple reflex, model-based, goal-based, utility based, learning agents) • Problem Solving & Search: Problem formulation, Uninformed search: BFS, DFS, Uniform Cost, IDS, Informed search: Greedy, A*, Heuristics: admissibility & consistency, Adversarial search: Minimax & Alpha-Beta pruning and constraint satisfaction problems (CSP) with backtracking (illustrated using the N-Queens problem). • Knowledge Representation & Logical Reasoning: Knowledge-based agents, Propositional logic & inference, First-Order Logic, Rule-based systems, Planning basics 	

(STRIPS concept)

- **Reasoning Under Uncertainty:** Probabilistic reasoning, Bayes theorem, Conditional independence, Bayesian Networks

UNIT II:

- **Introduction of Machine Learning:** Forms of learning (supervised, unsupervised, reinforcement), Parametric vs Nonparametric models, Bias–variance tradeoff, Overfitting & regularization, Gradient descent (intuitive)
- **Supervised Learning Models:** Classification vs Regression, k-NN, Decision Trees, Naive Bayes, SVM, Artificial Neural Networks:(single-layer & concept of deep learning, Ensemble methods & Boosting
- **Probabilistic & Latent Variable Models:** Statistical learning framework, Maximum Likelihood Estimation, Learning with complete data, Hidden variables, EM Algorithm, Hidden Markov Models
- **Unsupervised & Reinforcement Learning:** Concept of clustering, Association rule mining (Apriori concept), Reinforcement learning framework, Markov Decision Processes, Q-Learning (update rule intuition)
- **Generative AI and Responsible AI:** Introduction to Generative AI and LLMs, Prompt Engineering, Applications of Generative AI, Bias and Fairness in AI, Transparency, Accountability, and Human Oversight, Risks of Generative AI: Hallucinations, Deepfakes, and Misuse.
Demonstration of Zero-Shot, One-Shot, and Few-Shot Prompting Techniques Using Generative AI Tools, with Analysis of the Quality, Reliability, and Consistency of AI-Generated Responses.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

	Internal Examination: Continuous Evaluation – 20 marks	
	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
	Total 20	
11	REFERENCES: <ol style="list-style-type: none"> 1. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, 3rd Edition, Pearson, 2010. 2. Artificial Intelligence: Foundations of Computational Agents, David L Poole, Alan K. Mackworth, 2nd Edition, Cambridge University Press ,2017. 3. Artificial Intelligence, Kevin Knight and Elaine Rich, 3rd Edition, 2017 4. The Elements of Statistical Learning, Trevor Hastie, Robert Tibshirani and Jerome Friedman, Springer, 2013 	

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Cyber and Information Security

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Cyber and Information Security
Course Code: CHMCSV2

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Cyber & Information Security course provides students with an understanding of the fundamental principles and mechanisms used to protect digital information, communication systems, and computing infrastructure. It covers core security concepts, classical and modern cryptography, authentication methods, network security protocols, intrusion detection systems, malware threats, and firewall architectures.</p> <p>Relevance:</p> <p>In today’s digital era, cybersecurity plays a vital role in protecting data, privacy, and critical infrastructure. This course is highly relevant as organizations increasingly rely on secure communication, encryption, identity management, and network defense mechanisms to safeguard systems against evolving cyber threats.</p> <p>Usefulness:</p> <p>The course provides students with a strong foundation in encryption methods, authentication protocols, digital signatures, secure communication techniques, and network security mechanisms. It develops the skills needed to identify security vulnerabilities, apply appropriate safeguards, and design secure information systems.</p> <p>Application:</p> <p>Cyber and information security techniques are extensively used in banking, e-commerce, cloud computing, government services, defense, and enterprise networks. Security technologies including AES, RSA, SSL/TLS, IPsec, Kerberos, digital signatures, intrusion detection systems, and firewalls help ensure confidentiality, integrity, and secure access to digital resources.</p>

		<p>Interest:</p> <p>The subject is highly engaging as it explores real-world cyber threats, encryption techniques, malware analysis, and network defense mechanisms. Topics such as cryptographic algorithms, digital forensics fundamentals, ethical hacking concepts, and DDoS mitigation provide practical insights into modern cybersecurity challenges.</p> <p>Connection with Other Courses:</p> <p>The course complements core areas such as Computer Networks, Operating Systems, Discrete Mathematics, Cloud Computing, and Software Engineering by emphasizing their role in cybersecurity. It enhances interdisciplinary understanding necessary for the development of secure and reliable computing environments.</p> <p>Demand in the Industry:</p> <p>The demand for cybersecurity professionals is expanding significantly across sectors including IT services, financial technology, healthcare, telecommunications, defense, and cloud service providers. These organizations require skilled personnel in areas such as encryption, authentication, network security, intrusion detection, and firewall management to safeguard information systems and meet security and compliance requirements.</p> <p>Job Prospects:</p> <p>Graduates of this course are well-prepared for roles including cybersecurity analyst, network security engineer, information security officer, SOC analyst, penetration testing trainee, and security consultant. Further specialization can open opportunities in cryptography research, cloud security architecture, digital forensics, and security auditing.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations,

		Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To provide a strong foundation in cybersecurity concepts, including security services, attacks, threats, and defense mechanisms.</p> <p>CO(A) 2.To develop a comprehensive understanding of symmetric and asymmetric cryptography, along with key management techniques.</p> <p>CO(A) 3.To equip students with knowledge of authentication mechanisms, digital signatures, and secure communication protocols.</p> <p>CO(A) 4.To introduce network security technologies, intrusion detection systems, and techniques for detecting malware threats.</p> <p>CO(A) 5.To enable students to understand firewall architectures and the implementation of secure network infrastructures, along with the integration of Artificial Intelligence (AI) techniques in cybersecurity for enhanced threat detection and defense mechanisms.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Explain the core concepts of security architecture, cyber threats, attack models, and cryptographic principles.</p> <p>CO2: Apply symmetric and asymmetric cryptographic techniques, including AES, RSA, and Diffie–Hellman, to secure digital communication.</p> <p>CO3: Demonstrate the use of authentication mechanisms, digital signatures, hash functions, MACs, and Public Key Infrastructure (PKI) for ensuring data integrity and authenticity.</p> <p>CO4: Analyze network and web security protocols such as SSL/TLS, IPsec, Kerberos, and secure email protocols to protect communication systems.</p> <p>CO5: Identify and assess intrusion techniques, malware threats and firewall configurations for effective defense, incorporating Artificial Intelligence (AI)–based methods for threat detection, analysis, and mitigation in cybersecurity systems.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction: Security Trends, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms • Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Steganography, Block Cipher Principles, The Data Encryption Standard, The Strength of DES, AES (round details not expected), Multiple Encryption 	

	<p>and Triple DES, Block Cipher Modes of Operation, Stream Ciphers</p> <ul style="list-style-type: none"> • Public-Key Cryptography and RSA: Principles of Public-Key Cryptosystems, The RSA Algorithm • Key Management: Public-Key Cryptosystems, Key Management, Diffie-Hellman Key Exchange • Message Authentication and Hash Functions: Authentication Requirements, Authentication Functions, Message Authentication Codes, Hash Functions, Security of Hash Functions and Macs, Secure Hash Algorithm, HMAC <p>UNIT II:</p> <ul style="list-style-type: none"> • Digital Signatures and Authentication: Digital Signatures, Authentication Protocols, Digital Signature Standard • Authentication Applications: Kerberos, X.509 Authentication, Public-Key Infrastructure • Electronic Mail Security: Pretty Good Privacy, S/MIME • IP Security: Overview, Architecture, Authentication Header, Encapsulating Security Payload, Key Management • Web Security: Web Security Considerations, Secure Socket Layer and Transport Layer Security • Intrusion: Intruders, Intrusion Techniques, Intrusion Detection • Malicious Software: Viruses and Related Threats • Firewalls: Types of Firewalls • AI in Cybersecurity: Introduction to AI in cybersecurity, AI-based intrusion detection, machine learning for malware detection, AI-powered phishing detection, adversarial attacks and defense mechanisms, and overview of AI-enabled cybersecurity tools and applications.
<p>10</p>	<p style="text-align: center;">Scheme of Examination and Assessment Pattern</p> <p style="text-align: center;">Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 01:00 hours</p> <p style="text-align: center;">Format of Question Paper</p>

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Cryptography and Network Security: Principles and Practice 7th edition, William Stallings, Pearson 2. Cryptography and Network Security, 2nd edition, Behrouz A Fourouzan, Debdeep Mukhopadhyay, TMH. 3. Atul Kahate, “Cryptography and Network Security”, Tata McGraw-Hill.
-----------	---

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: IKS in Computational Systems

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: IKS in Computational Systems

Course Code: CHMCSV3

Sr.No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The IKS in Computational Systems – Research Based Project course explores classical Indian Knowledge Systems (IKS) through the lens of modern computational thinking. It examines how śāstra-based formalization, Pāṇini’s generative grammar, Piṅgala’s combinatorics, Nyāya logic, Ayurvedic classification, Arthaśāstra cryptography, and other knowledge traditions embody structured, rule-based, and algorithmic principles analogous to contemporary Computer Science concepts.</p> <p>Through guided research, literature review, conceptual modelling, comparative analysis, case studies, and computational investigations, students explore the relevance of these traditional knowledge systems in the context of modern computing paradigms.</p> <p>Relevance:</p> <p>This course is highly relevant in the context of interdisciplinary education envisioned under NEP 2020, linking traditional knowledge frameworks with modern computational theory. It demonstrates that formal systems, symbolic encoding, logical inference, structured reasoning, classification mechanisms, and algorithmic processes have deep historical roots aligned with contemporary developments in Artificial Intelligence, Natural Language Processing, Knowledge Representation, Formal Grammars, Combinatorics, Data Analytics, and Cyber Security.</p> <p>Usefulness:</p> <p>The course develops conceptual clarity in knowledge representation, rule-based reasoning, generative systems, symbolic abstraction, inference validation, classification methodologies, and computational thinking. It strengthens</p>

analytical and research skills by encouraging students to investigate classical structures such as pramāṇa theory, prastāra enumeration, generative grammar, and syllogistic reasoning through modern computational frameworks.

The project-based approach further develops competencies in literature review, academic writing, research methodology, critical analysis, scholarly communication, and interdisciplinary problem solving.

Application:

The computational parallels of śāstra methodology, Pāṇinian grammar, Nyāya logic, Ayurvedic reasoning, and Arthaśāstra cryptography have applications in formal language theory, natural language processing, expert systems, decision trees, inference engines, knowledge graphs, explainable AI, secure communication systems, and computational modeling.

Students gain insight into how structured reasoning and knowledge organization models can inform AI system design, feature engineering, classification systems, semantic technologies, and algorithm development.

Interest

The course is intellectually stimulating as it reveals algorithmic, logical, linguistic, mathematical, and organizational sophistication embedded in Indian knowledge traditions. Discovering binary encoding in Piṅgala, generative grammar in Pāṇini, inference structures in Nyāya, multi-valued reasoning in Jain logic, knowledge representation in Navya-Nyāya, computational astronomy, and coded communication in Arthaśāstra creates a unique synthesis of traditional scholarship and modern Computer Science.

The research-based format encourages students to formulate research questions, investigate classical sources, develop conceptual models, and generate innovative interpretations

of computational ideas rooted in Indian Knowledge Systems.

Connection with Other Courses:

This course connects strongly with several core areas of Computer Science and Information Technology. It complements Artificial Intelligence through concepts of knowledge representation, reasoning, and explainable AI; Machine Learning through classification and feature engineering; and Natural Language Processing through grammar, parsing, and semantic analysis. The course also relates to Formal Languages and Automata Theory by exploring formal grammars and rule-based systems, and to Design and Analysis of Algorithms through recursive and combinatorial methods. Further, it supports learning in Data Science and Analytics through classification and decision models, Knowledge Engineering and Semantic Technologies through structured knowledge representation, and Cyber Security through cryptographic principles and secure communication models. The research-oriented approach of the course also aligns with Research Methodology and Theory of Computation. Overall, the course enriches computational disciplines by providing historical, philosophical, and interdisciplinary perspectives that connect traditional Indian knowledge systems with modern computing paradigms.

Demand in the industry:

Modern fields such as Artificial Intelligence, Natural Language Processing, Knowledge Engineering, Data Analytics, Semantic Computing, Explainable AI, and Cyber Security rely heavily on rule-based systems, logical inference, structured knowledge representation, and computational reasoning. The course highlights foundational principles behind these technologies through classical knowledge frameworks while simultaneously developing research and analytical skills valued across industry and academia.

		<p>Job Prospects:</p> <p>The concepts, analytical abilities, and research skills developed through this course support career opportunities in emerging areas such as Artificial Intelligence, Data Science and Analytics, Computational Linguistics, Natural Language Processing, Knowledge Engineering, Semantic Technologies, Cyber Security, Research and Development, and Digital Humanities. By integrating traditional knowledge systems with modern computational perspectives, the course equips students with interdisciplinary problem-solving and critical thinking skills that are increasingly valued in both industry and academia. The analytical, computational, and research-oriented perspective gained through the course also provides a strong foundation for higher studies and advanced research in Computer Science, Artificial Intelligence, Computational Linguistics, Information Systems, Digital Humanities, and Indian Knowledge Systems.</p>
2	Vertical	Major - IKS
3	Type	Theory
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Explore computational principles embedded in Indian Knowledge Systems through systematic research and scholarly inquiry.</p> <p>CO(A) 2. Investigate the relationship between traditional Indian knowledge frameworks and modern Computer Science concepts such as algorithms, logic, knowledge representation, artificial intelligence, and cryptography.</p> <p>CO(A) 3. Develop competencies in literature review, research methodology, analytical thinking, academic writing, and presentation of research findings.</p> <p>CO(A) 4. Analyze and compare classical Indian knowledge systems with contemporary computational models using interdisciplinary approaches.</p> <p>CO(A) 5. Design and execute a research project that demonstrates the relevance and application of Indian Knowledge Systems in Computational Sciences.</p>	

<p>8</p>	<p>Course Outcomes:</p> <p>After successful completion of this course, students will be able to:</p> <p>CO1: Conduct systematic literature reviews and identify research problems related to Indian Knowledge Systems and Computational Sciences.</p> <p>CO2: Analyze computational, logical, linguistic, and mathematical concepts present in classical Indian knowledge traditions.</p> <p>CO3: Apply research methodologies and analytical techniques to investigate selected IKS themes from a computational perspective.</p> <p>CO4: Develop and present well-structured research reports, case studies, conceptual models, and seminar presentations.</p> <p>CO5: Demonstrate interdisciplinary understanding and critical thinking by establishing connections between Indian Knowledge Systems and modern computational technologies.</p>
<p>9</p>	<p style="text-align: center;">Syllabus Project Themes / Research Areas</p> <p>Students shall undertake one research project from any one of the following themes:</p> <p>1. Śāstra Methodology and Knowledge Formalization Topics</p> <ul style="list-style-type: none"> • Śāstra as a formal knowledge system • Sūtra methodology and symbolic representation • Pramāṇa-based knowledge validation • Structured reasoning frameworks <p>Modern CS Connection Knowledge Representation, Formal Specification, Symbolic Systems</p> <p>2. Piṅgala’s Chandaḥśāstra and Binary Encoding Topics</p> <ul style="list-style-type: none"> • Laghu–Guru representation • Binary structures • Prastāra generation • Meru-Prastāra <p>Modern CS Connection Binary Systems, Combinatorics, Recursive Algorithms</p> <p>3. Pāṇini’s Aṣṭādhyāyī and Formal Grammar Topics</p> <ul style="list-style-type: none"> • Rule-based language generation • Meta-rules and precedence • Context-sensitive operations <p>Modern CS Connection Formal Grammars, Parsing, Natural Language Processing (NLP)</p> <p>4. Nyāya Logic and Inference Systems Topics</p> <ul style="list-style-type: none"> • Five-member syllogism • Anumāna

- Debate and validation methods

Modern CS Connection

Logic Systems, Inference Engines, Explainable AI

5. Ayurvedic Classification and Expert Systems

Topics

- Tridoṣa framework
- Diagnostic reasoning
- Feature mapping

Modern CS Connection

Expert Systems, Decision Trees, Classification Models

6. Arthaśāstra Cryptography and Secure Communication

Topics

- Secret communication
- Coding techniques
- Information protection

Modern CS Connection

Cryptography, Secure Communication, Cyber Security

7. Navya-Nyāya and Knowledge Representation

Topics

- Semantic precision
- Relational structures
- Knowledge organization

Modern CS Connection

Ontologies, Knowledge Graphs, Semantic Networks

8. Jain Logic and Multi-Valued Reasoning

Topics

- Syādvāda
- Anekāntavāda
- Context-dependent truth

Modern CS Connection

Fuzzy Logic, Uncertainty Modeling, AI Reasoning

9. Indian Mathematics and Algorithmic Computation

Topics

- Mathematical algorithms
- Recursive methods
- Numerical techniques

Modern CS Connection

Algorithms, Numerical Computing, Complexity Analysis

10. Sanskrit and Natural Language Processing

Topics

- Morphological analysis
- Sandhi processing
- Semantic structures

Modern CS Connection

Natural Language Processing, Machine Translation, Computational Linguistics

11. Temple Architecture and Computational Geometry

Topics

- Geometric design principles
- Symmetry and proportions
- Spatial structures

Modern CS Connection

Computational Geometry, CAD Systems, Computer Graphics

12. Indian Astronomical Computation and Predictive Modeling

Topics

- Astronomical calculations
- Calendar systems
- Planetary computations

Modern CS Connection

Simulation, Predictive Modeling, Scientific Computing

13. Kautilya's Administrative Systems and Decision Analytics

Topics

- Information management
- Strategic planning
- Resource allocation

Modern CS Connection

Decision Support Systems, Operations Research, Analytics

14. Indian Epistemology and Artificial Intelligence

Topics

- Pramāṇa theory
- Knowledge acquisition
- Validation mechanisms

Modern CS Connection

AI Knowledge Systems, Explainable AI, Cognitive Computing

15. Vedic Mathematics and Computational Efficiency

Topics

- Mental computation techniques
- Mathematical shortcuts
- Pattern-based problem solving

Modern CS Connection

Algorithm Optimization, Computational Efficiency, Fast Computation Methods

Students may choose any one theme and formulate a specific research problem in consultation with the course instructor. The project may be theoretical, analytical, comparative, conceptual, computational, experimental, or prototype-based depending upon the nature of the selected topic. If required faculty members will allot Projects topics in addition to the above list that are relevant and important to the core Subject. The Project may be taken individually or in a group of students with proper guidance and prior sanction from the faculty.

Guidelines for Project Work

Students shall undertake a research-based project on any one of the prescribed themes. The project should demonstrate the student's ability to investigate, analyze, interpret, and present

the computational aspects of Indian Knowledge Systems using appropriate research methods.

Suggested Approaches

A. Literature-Based Research Study

- Systematic review of classical texts, books, journal articles, and research papers.
- Comparative analysis of traditional concepts and modern computational theories.
- Identification of computational principles embedded in Indian Knowledge Systems.

B. Conceptual Modeling

- Development of logical, mathematical, linguistic, or computational models based on IKS concepts.
- Creation of flowcharts, frameworks, ontologies, decision models, knowledge graphs, or process models.
- Mapping of traditional knowledge structures to modern Computer Science concepts.

C. Experimental or Computational Study

- Simulation, algorithmic implementation, data analysis, computational verification, or prototype development.
- Development of small software tools, demonstrations, proof-of-concept applications, or analytical models where applicable.

D. Case Study and Comparative Analysis

- Investigation of selected texts, systems, methodologies, or applications.
- Comparative study between Indian Knowledge Systems and contemporary computational approaches.

Nature of Deliverables

The project deliverable may consist of one or more of the following:

- Research Report
- Review Paper
- Comparative Study
- Conceptual Framework
- Knowledge Representation Model
- Ontology or Knowledge Graph
- Algorithm Design
- Flowchart-Based Model
- Computational Analysis
- Software Prototype
- Simulation Study
- Case Study Report

Note: Since the course encompasses diverse themes within Indian Knowledge Systems, not all projects are expected to result in software development or programming-based outcomes.

For example:

- Projects on Pāṇini's Aṣṭādhyāyī, Nyāya Logic, Jain Logic, Śāstra Methodology, or Indian Epistemology may focus primarily on conceptual modeling, formal analysis,

logical structures, knowledge representation, comparative studies, or framework development.

- Projects on Piṅgala’s Combinatorics, Sanskrit and NLP, Cryptography in Arthaśāstra, Indian Mathematics, or Astronomical Computation may additionally include computational implementation, simulation, algorithm development, or prototype creation.

Therefore, for evaluation purposes, the term “Prototype Development / Experimental Implementation / Conceptual Model Demonstration” shall include software prototypes, algorithmic implementations, simulations, conceptual frameworks, ontologies, decision models, knowledge graphs, formal representations, or any other research artifact appropriate to the chosen topic.

The emphasis of the course is on originality, analytical rigor, interdisciplinary understanding, and the ability to relate Indian Knowledge Systems to modern computational sciences.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End

External - 30 marks

The External Assessment shall evaluate the final research work, outcomes, findings, report quality, and viva voce examination.

	Assessment Component	Marks
1.	Prototype Development / Experimental Implementation / Conceptual Model Demonstration	10
2.	Analysis, Interpretation, and Discussion of Results / Findings	10
3.	Final Research Report / Research Paper	05
4.	Viva Voce Examination	05
	Total	30

Guidelines for External Assessment

Prototype Development / Experimental Implementation / Conceptual Model Demonstration (10 Marks)

- Development of software prototype, simulation, algorithm, conceptual framework, ontology, knowledge graph, decision model, comparative model, or other project artifact appropriate to the selected theme
- Originality and relevance of the work
- Technical and conceptual quality

Analysis, Interpretation, and Discussion of Results / Findings (10 Marks)

- Quality of analysis
- Interpretation of observations and findings

- Computational relevance and insights
- Critical discussion and conclusions

Final Research Report / Research Paper (5 Marks)

The report should include:

- Title and Abstract
- Objectives
- Literature Review
- Methodology
- Analysis and Findings
- Conclusion
- References

Viva Voce Examination (5 Marks)

- Understanding of the selected topic
- Knowledge of research methodology
- Ability to explain findings and conclusions
- Response to questions from examiners

Internal Examination: Continuous Evaluation- 20 marks

The Internal Assessment shall evaluate the student's ability to identify a research problem, conduct literature review, plan the project, and communicate progress during the semester.

	Assessment Component	Marks
1.	Research Problem Identification, Objectives Formulation, and Literature Review	10
2.	Internal Seminar / Progress Review (Problem Understanding, Methodology, Presentation Skills, Communication and Discussion)	10
	Total	20

Guidelines for Internal Assessment

Research Problem Identification and Literature Review (10 Marks)

- Selection and justification of research topic
- Clarity of research objectives
- Identification of research gap/problem statement
- Review of relevant literature and sources
- Understanding of IKS and computational perspectives

Internal Seminar / Progress Review (10 Marks)

- Project planning and methodology
- Progress made during the semester

	<ul style="list-style-type: none"> • Presentation skills and communication • Ability to answer questions and justify approach • Active participation in review discussions
<p>11</p>	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Computing Science in Ancient India, T.R.N. Rao/ Subhash Kak 2. The Mathematics of India: Concepts, Methods, Connections P. P. Divakaran, Hindustan Book Agency, 2018 3. Pāṇini: Background and introduction, George Cardona, Motilal Banarsidass, 1988 4. Nyaya-Vaisesika (A History of Indian Literature) B. K. Matilal, Harrassowitz, 1977 5. The Computation Meme: Explorations in Indic Computational Thinking K. Gopinath & Shailaja D. Sharma, Indian Institute of Science (IISc), Bengaluru, 2022 6. Indian Mathematics Engaging the World from Ancient to Modern Times G. G. Joseph, 2016 7. Research papers, conference proceedings, theses, and digital archives related to Indian Knowledge Systems and Computational Sciences. 8. Publications from the Indian Knowledge Systems Division, Ministry of Education, Government of India.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Computer Science Practical 5

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Computer Science Practical 5
Course Code: CHMCSV4

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>This practical course provides hands-on training in Artificial Intelligence and Cyber & Information Security through structured laboratory exercises. Students implement theoretical concepts, develop practical solutions, and perform experimentation, performance evaluation, comparative analysis, and real-world system implementation.</p> <p>Relevance:</p> <p>This course is highly relevant as it integrates the interdisciplinary domains of Artificial Intelligence and Cyber & Information Security through practical application. It enables students to implement and analyze AI algorithms, machine learning techniques, cryptographic methods, and network security mechanisms in real-world scenarios. By combining theoretical concepts with hands-on laboratory exercises, the course bridges the gap between academic knowledge and practical proficiency.</p> <p>Usefulness:</p> <p>Through hands-on laboratory exercises, students gain practical experience in implementing algorithms, working with datasets, evaluating machine learning models, developing cryptographic applications, and configuring security mechanisms. The course strengthens debugging, analytical thinking, algorithm comparison, performance evaluation, and secure system implementation skills, preparing students for real-world and industry-oriented challenges.</p> <p>Application:</p> <p>The AI practicals cover applications such as intelligent search, predictive analytics, classification, neural networks, and data-driven decision-making. The Cyber & Information Security practicals focus on secure communication, cryptographic techniques, digital</p>

signatures, firewall configuration, intrusion detection, and secure web technologies, providing skills applicable to modern enterprise and cloud-based environments.

Interest:

Student engagement is enhanced through implementation-based learning, algorithm benchmarking, model visualization, cryptography experiments, and network security configuration exercises. Practical exposure to AI frameworks (e.g., TensorFlow and OpenAI APIs) and security technologies such as SSL/TLS, intrusion detection systems (IDS), and firewall management provides a technically rigorous and application-oriented learning environment.

Connection with Other Courses:

This course builds upon concepts from Artificial Intelligence, Cyber & Information Security, Data Structures, Computer Networks, Operating Systems, and Cloud Computing. It integrates theoretical knowledge with practical implementation, enhancing students' understanding of algorithms, secure computing, and system design while fostering interdisciplinary problem-solving skills.

Demand in the Industry:

The industry demands professionals who can effectively implement and deploy AI and cybersecurity solutions in addition to understanding their underlying concepts. This course equips students with practical skills in machine learning, cryptographic techniques, secure communication protocols, intrusion detection, and network security, preparing them for roles in IT, cybersecurity, fintech, and cloud computing environments.

Job Prospects:

Upon completing this practical course, students are well prepared for entry-level roles such as AI/ML intern, data analyst, cybersecurity analyst, SOC analyst, security operations trainee, and junior DevOps engineer. The hands-on experience gained through the course also provides a strong foundation for advanced careers in

		machine learning, cybersecurity, system administration, and security consulting.
2	Vertical 1	Major (Mandatory)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Develop practical proficiency in implementing AI search algorithms and machine learning models for solving real-world problems.</p> <p>CO(A) 2. Train, evaluate, and compare supervised learning, ensemble learning, and URL-based phishing detection models using appropriate performance metrics.</p> <p>CO(A) 3. Acquire hands-on experience in implementing classical and modern cryptographic techniques to ensure secure communication and data protection.</p> <p>CO(A) 4. Apply cybersecurity concepts through secure communication protocols, intrusion detection mechanisms, firewall configuration, and phishing website analysis.</p> <p>CO(A) 5. Strengthen analytical, debugging, and problem-solving skills through system experimentation, performance evaluation, comparative analysis, and AI-assisted validation.</p>	
8	<p>Course Outcomes: -</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Implement and compare AI search algorithms and supervised machine learning models, evaluating their performance using appropriate metrics and analytical techniques.</p> <p>CO2: Design, train, and analyze machine learning models including decision trees, neural networks, support vector machines, ensemble methods, association rule mining algorithms, and URL-based phishing detection systems.</p> <p>CO3: Implement and validate cryptographic techniques such as RSA, Diffie–Hellman, message authentication codes (MACs), and digital signatures to ensure secure communication.</p> <p>CO4: Configure and analyze network security mechanisms including SSL/TLS, IPsec, intrusion detection systems, firewall rules, and phishing detection techniques.</p> <p>CO5: Evaluate the effectiveness and performance of AI and cybersecurity solutions through practical implementation and analysis.</p>	

Syllabus

UNIT I:

Practical 1: Breadth First Search & Iterative Depth First Search

- Implement the Breadth First Search algorithm to solve a given problem.
- Implement the Iterative Depth First Search algorithm to solve the same problem.
- Compare the performance and efficiency of both algorithms.

Practical 2: A* Search and Recursive Best-First Search

- Implement the A* Search algorithm for solving a pathfinding problem.
- Implement the Recursive Best-First Search algorithm for the same problem.
- Compare the performance and effectiveness of both algorithms.

Practical 3: Backtracking-Based Search

- Implement the Backtracking-Based Search algorithm to solve a constraint satisfaction problem (e.g., the 8-Queens problem).
- Generate a valid solution by systematically exploring the search space and backtracking whenever constraints are violated.
- Analyze the search process, backtracking steps, and overall effectiveness of the algorithm in finding the solution.

Practical 4: Decision Tree Learning

- Implement the Decision Tree Learning algorithm to build a decision tree for a given dataset.
- Evaluate the accuracy and effectiveness of the decision tree on test data.
- Visualize and interpret the generated decision tree.

Practical 5: Feed Forward Backpropagation Neural Network

- Implement the Feed Forward Backpropagation algorithm to train a neural network.

- Use a given dataset to train the neural network for a specific task.
- Evaluate the performance of the trained network on test data

Practical 6:

Support Vector Machines (SVM)

- Implement the SVM algorithm for binary classification.
- Train an SVM model using a given dataset and optimize its parameters.
- Evaluate the performance of the SVM model on test data and analyze the results.

Practical 7:

Adaboost Ensemble Learning:

- Implement the Adaboost algorithm to create an ensemble of weak classifiers.
- Train the ensemble model on a given dataset and evaluate its performance.
- Compare the results with individual weak classifiers

Practical 8:

Naive Bayes' Classifier

- Implement the Naive Bayes' algorithm for classification.
- Train a Naive Bayes' model using a given dataset and calculate class probabilities.
- Evaluate the accuracy of the model on test data and analyze the results

Practical 9:

K-Nearest Neighbors (K-NN)

- Implement the K-NN algorithm for classification or regression.
- Apply the K-NN algorithm to a given dataset and predict the class or value for test data.
- Evaluate the accuracy or error of the predictions and analyze the results.

Practical 10:

Demo of OpenAI/TensorFlow Tools

- Explore and experiment with OpenAI or TensorFlow tools and libraries.
- Perform a demonstration or mini-project showcasing the capabilities of the tools.
- Discuss and present the findings and potential applications

Unit-II

Practical 1:

Implementing Substitution and Transposition Ciphers:

- Design and implement algorithms to encrypt and decrypt messages using classical substitution and transposition techniques.

- | | |
|--|---|
| | <p>Practical 2: RSA Encryption and Decryption:</p> <ul style="list-style-type: none">• Implement the RSA algorithm for public-key encryption and decryption, and explore its properties and security considerations. |
| | <p>Practical 3: Message Authentication Codes:</p> <ul style="list-style-type: none">• Implement algorithms to generate and verify message authentication codes (MACs) for ensuring data integrity and authenticity. |
| | <p>Practical 4: Digital Signatures:</p> <ul style="list-style-type: none">• Implement digital signature algorithms such as RSA-based signatures, and verify the integrity and authenticity of digitally signed messages. |
| | <p>Practical 5: Key Exchange using Diffie-Hellman:</p> <ul style="list-style-type: none">• Implement the Diffie-Hellman key exchange algorithm to securely exchange keys between two entities over an insecure network. |
| | <p>Practical 6: IP Security (IPsec) Configuration:</p> <ul style="list-style-type: none">• Configure IPsec on network devices to provide secure communication and protect against unauthorized access and attacks. |
| | <p>Practical 7: Web Security with SSL/TLS:</p> <ul style="list-style-type: none">• Configure and implement secure web communication using SSL/TLS protocols, including certificate management and secure session establishment. |
| | <p>Practical 8: Intrusion Detection System:</p> <ul style="list-style-type: none">• Set up and configure an intrusion detection system (IDS) to monitor network traffic and detect potential security breaches or malicious activities. |
| | <p>Practical 9: Firewall Configuration and Rule-based Filtering:</p> <ul style="list-style-type: none">• Configure and test firewall rules to control network traffic, filter packets based on specified criteria, and protect network resources from unauthorized access. |
| | <p>Practical 10: Phishing Website Detection Using ML:</p> <ul style="list-style-type: none">• Build a machine learning model to detect phishing websites using URL features and validate the prediction with a generative AI tool. |

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, 3rd Edition, Pearson, 2010.
2. Cryptography and Network Security: Principles and Practice 7th edition, William Stallings, Pearson
3. Artificial Intelligence: Foundations of Computational Agents, David L Poole, Alan K. Mackworth, 2nd Edition, Cambridge University Press ,2017.
4. Artificial Intelligence, Kevin Knight and Elaine Rich, 3rd Edition, 2017
5. The Elements of Statistical Learning, Trevor Hastie, Robert Tibshirani and Jerome Friedman, Springer, 2013
6. Cryptography and Network Security, 2nd edition, Behrouz A Fourouzan, Debdeep Mukhopadhyay, TMH.
7. Atul Kahate, "Cryptography and Network Security", Tata McGraw-Hill.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Mini Project – I

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Mini Project – I (Real-World Application Development)

Course Code: CHMCSV5

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>Mini Project – I is a structured, development-oriented course that enables students to design and implement a complete real-world software solution. The course guides students through problem identification, requirement engineering, system modeling, architecture design, implementation, testing, and deployment. It emphasizes professional documentation, structured development practices, and industry-standard tools.</p> <p>Relevance:</p> <p>This course is highly relevant as it simulates an industry style software development lifecycle from concept to deployment. Students experience real-world project constraints, stakeholder analysis, SDLC planning, and cloud deployment, preparing them for professional software development environments.</p> <p>Usefulness:</p> <p>The course develops practical competencies in requirement analysis, UML modeling, backend–frontend integration, database design, testing strategies, and version control using GitHub. It strengthens documentation skills, structured thinking, and collaborative development—essential capabilities for modern software engineers.</p> <p>Application:</p> <p>Students design solutions for real-world industry, social, or institutional problems such as service portals, management systems, automation tools, or mobile/web applications. The structured implementation and deployment process mirrors real software product development workflows used in startups, enterprises, and IT service organizations.</p> <p>Interest:</p> <p>The course generates strong engagement because students build a tangible, working application that solves a real problem. The opportunity to design architecture, implement features, deploy to cloud or mobile platforms, and demonstrate the final product</p>

		<p>makes the experience practically rewarding and professionally motivating.</p> <p>Connection with Other Courses:</p> <p>Mini Project – I integrates knowledge from Software Engineering, Database Management Systems, Web Development, Mobile Application Development, Cloud Computing, Cyber Security, and Data Structures. It acts as a capstone-style integrative experience that consolidates prior learning into a unified application.</p> <p>Demand in the Industry:</p> <p>Industry demands graduates who can independently conceptualize, design, develop, test, and deploy scalable applications. Skills such as requirement engineering, SDLC planning, UML modeling, GitHub version control, cloud deployment, and security validation are highly valued in software development, DevOps, and startup ecosystems.</p> <p>Job Prospects:</p> <p>Completion of this course strengthens readiness for roles such as software developer trainee, full-stack developer intern, backend/frontend developer, DevOps trainee, or application support engineer. A well-executed mini-project also enhances employability by serving as a portfolio project during placements.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To enable students to identify and analyze real-world problems and conduct feasibility studies.</p> <p>CO(A) 2. To develop skills in requirement engineering, UML-based system modeling, and architecture design.</p> <p>CO(A) 3. To provide hands-on experience in application development, testing, and deployment.</p> <p>CO(A) 4. To inculcate best practices in version control, documentation, and project management.</p>	

	<p align="center">CO(A) 5. To prepare students for professional software development environments through structured SDLC implementation.</p>
<p align="center">8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Identify and analyse a real-world problem and prepare structured project documentation including SRS and feasibility reports.</p> <p>CO2: Design complete system models using UML diagrams and develop suitable frontend, backend, and database architectures.</p> <p>CO3: Implement and integrate application components with authentication, validation, and error handling mechanisms.</p> <p>CO4: Test and deploy applications using systematic testing approaches and version control tools such as GitHub.</p> <p>CO5: Evaluate and present the developed solution through technical documentation, demonstration, and performance/security validation.</p>
<p align="center">9</p>	<p align="center">Syllabus</p> <p>UNIT I:</p> <p>Problem Identification, Requirement Engineering & System Design Phase</p> <p>Problem Identification & Feasibility Study: Identification of a real-world problem (industry / social / institutional), Problem justification and scope definition, Stakeholder identification, Feasibility analysis (technical, economic, operational)</p> <p>Requirement Engineering: Functional requirements specification, Non-functional requirements, Use-case analysis, Requirement prioritization, Constraints and assumptions</p> <p>Software Development Life Cycle (SDLC) Planning: Selection of SDLC model, Work Breakdown Structure (WBS), Project timeline (Gantt Chart), Resource planning</p> <p>System Modeling using UML: Use Case Diagram, Class Diagram, Sequence Diagram, Activity Diagram, ER Diagram, Deployment Diagram</p> <p>System Architecture Design: Frontend architecture, Backend architecture, Database schema design, API structure, Security considerations</p> <p><u>Documentation Deliverables at the End of Unit 1</u></p> <ul style="list-style-type: none"> • Project Proposal • SRS Document • Complete UML Set • Architecture Design Document <p>UNIT II:</p> <p>Implementation, Testing, Deployment & Evaluation Phase</p>

Application Development: Frontend implementation, Backend implementation, Database integration, Authentication & validation, Error handling
Integration & System Testing: Unit testing, Black-box testing, Integration testing, Test case preparation, Bug tracking
Deployment: Cloud deployment / Local hosting, APK build, Server configuration (if applicable), Version control using GitHub (Mandatory)
Performance & Security Testing: Basic load testing, Input validation checks, Security validation
Final Documentation: Technical Report, User Manual, Screenshots, Source Code Documentation

Final Deliverables at the End of Module 2

- Working Application
- GitHub Repository
- Final Report
- Presentation & Demonstration

10

Evaluation for Mini Project

Paper – 50 Marks

Mini Project – I

Internal Evaluation (To be assessed by the Project Guide)	Component	Marks
		Problem Identification & Project Proposal
	System Design (SRS, UML Diagrams, Architecture)	5
	Development Progress & Code Review	5
	Internal Presentation / Review	5
External Evaluation (To be assessed by External Examiner)	Component	Marks
	Working Application Demonstration	10
	Technical Design & Implementation	10
	Project Report Evaluation	5
	Viva Voce	5

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Wireless & Sensor Networks

Vertical - 1

Major (Elective) Course - 2 Credits

**with effect from
Academic Year 2026-2027**

Title: Wireless & Sensor Networks
Course Code: CHMCSV6

Sr.No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Wireless Sensor Networks (WSNs) course introduces students to the fundamental concepts, architecture, protocols, and communication mechanisms of sensor-based wireless systems. The course covers sensor node technologies, network architecture, WSN operating systems, data aggregation techniques, ad-hoc networking, MAC and routing protocols, transport mechanisms, and wireless communication fundamentals. It also includes the study of mobile communication technologies, telecommunication systems, and satellite-based communication networks.</p> <p>Relevance:</p> <p>Wireless Sensor Networks play a vital role in modern communication infrastructures and emerging technologies such as the Internet of Things (IoT), smart cities, industrial automation, environmental monitoring, and intelligent transportation systems. This course provides the theoretical and practical foundation required to understand, design, deploy, and manage wireless sensor-based communication systems in diverse real-world applications.</p> <p>Usefulness:</p> <p>The course enables students to understand energy-efficient communication, data aggregation, wireless transmission principles, routing challenges, network optimization, and security considerations in sensor networks. It develops analytical and problem-solving skills needed to design reliable, scalable, and efficient wireless communication solutions for resource-constrained environments.</p> <p>Application:</p>

Wireless Sensor Network technologies are widely applied in smart agriculture, healthcare monitoring, industrial automation, environmental sensing, disaster management, military surveillance, smart homes, and smart grid systems. Concepts related to IEEE 802.15.4, mobile communication systems, DECT, UMTS, and satellite networks support applications in telecommunications, IoT ecosystems, and large-scale wireless communication infrastructures.

Interest:

The course is intellectually engaging as it explores how distributed sensor nodes communicate, coordinate, and exchange information in dynamic wireless environments. Topics such as energy-aware networking, data aggregation, ad-hoc communication, wireless transmission, mobile handover, and satellite routing provide valuable insights into the design of modern communication systems and emerging networking technologies.

Connection with Other Courses:

This course complements Computer Networks, Operating Systems, Embedded Systems, Internet of Things (IoT), Cyber Security, Mobile Computing, and Cloud Computing. It integrates concepts related to network protocols, system design, wireless communication, distributed computing, and secure information exchange across interconnected devices.

Demand in the industry:

The increasing adoption of IoT solutions, smart infrastructure, industrial automation, and next-generation communication networks has created significant demand for professionals skilled in wireless networking and sensor technologies. Industries including telecommunications, manufacturing, healthcare, defense, automotive systems, and satellite communication actively seek expertise in wireless network design, deployment, and optimization.

Job Prospects:

		Students can pursue careers as Wireless Network Engineers, IoT Developers, Telecom Network Analysts, Embedded Systems Engineers, Communication Systems Engineers, Network Planning Engineers, and Wireless Solution Architects. Advanced expertise in wireless and sensor networks can also lead to opportunities in research and development, satellite communications, next-generation mobile networks, and intelligent connected systems.
2	Vertical	Major (Elective)
3	Type	Theory
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. Understand the fundamentals, architecture, applications, and design principles of Wireless Sensor Networks (WSNs).</p> <p>CO(A) 2. Explain WSN operating systems, data management techniques, ad-hoc networks, and communication protocols including MAC, routing, and transport layers.</p> <p>CO(A) 3. Describe wireless transmission concepts such as signal propagation, antennas, modulation, multiplexing, spread spectrum, and cellular communication.</p> <p>CO(A) 4. Analyze mobile communication and satellite systems, including DECT, UMTS, IMT-2000, localization, handover, and security mechanisms.</p> <p>CO(A) 5. Apply WSN concepts and tools to design, evaluate, and optimize sensor network solutions for real-world applications.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students will be able to:</p> <p>CO1: Explain the fundamental concepts, architecture, operating environments, and design objectives of Wireless Sensor Networks (WSNs), and apply them to develop effective WSN solutions.</p> <p>CO2: Evaluate and compare WSN communication protocols, including MAC, routing, and transport mechanisms, to improve network efficiency, reliability, and performance.</p> <p>CO3: Analyze wireless communication principles such as radio frequencies, signal propagation, antennas, modulation, multiplexing, and wireless channel characteristics</p>	

to assess their impact on WSN operation.

CO4:Examine telecommunication, mobile communication, satellite, and broadcast systems, and analyze their role in supporting wireless networking and real-world WSN deployments.

CO5:Design, simulate, and evaluate WSN-based applications by applying theoretical concepts, data management techniques, and practical implementation methodologies.

9	<h2>Syllabus</h2> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction and Overview of Wireless Sensor Networks: Fundamental concepts and architectural components of sensor networks, Advantages and applications of Wireless Sensor Networks (WSNs), Sensor node technology and taxonomy, Wireless network operating environments, Radio technologies, Network architecture and optimization, Optimization goals, Performance metrics (figures of merit), Design principles for WSNs, Service interfaces of WSNs, Gateway concepts for WSN deployment and management. • Wireless Sensor Network Operating Systems and Ad-hoc Networks: Overview of operating systems developed for Wireless Sensor Networks (WSNs), Architecture and features of WSN operating systems, Examples of WSN operating systems, Data collection, aggregation and fusion techniques, Data management for efficient communication, Characteristics of ad-hoc networks in WSNs, Dynamic topology and resource constraints, Energy efficiency strategies, Security and privacy issues in ad-hoc networks. • Medium Access Control (MAC) in WSN: Fundamentals and overview of MAC protocols for sensor networks, Sensor-MAC (S-MAC) protocol and case study. • Routing in WSN: Routing challenges and design issues in Wireless Sensor Networks, Routing strategies in WSNs, IEEE 802.15.4 LR-WPAN standard (case study). • Transport Layer and Middleware in WSN: Traditional transport control protocols, Transport protocol design issues in WSNs, WSN middleware architecture. <p>UNIT-II:</p> <ul style="list-style-type: none"> • Wireless Transmission Fundamentals: Frequency for radio transmission, Signals, Antennas, Signal propagation, Wireless channel impairments (attenuation, fading, interference and noise), Multiplexing techniques, Modulation techniques, Spread spectrum technologies, Cellular systems, Energy efficiency in wireless communication. • Telecommunication and Mobile Communication Systems: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover mechanisms, Security, Data services. • Advanced Mobile Communication Systems: DECT – system architecture and protocol architecture, TETRA, UMTS, IMT-2000, Evolution of mobile communication standards.
---	--

- **Satellite and Broadcast Systems:** History of satellite systems, Applications of satellite systems, Satellite fundamentals (GEO, LEO and MEO orbits), Routing in satellite systems, Localization, Handover mechanisms, Broadcast systems overview.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation- 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
	Total	20

11

REFERENCES:

1. Wireless Sensor Networks: Technology, Protocols, and Applications – Kazem Sohraby, Daniel Minoli, and Taieb Znati, John Wiley & Sons, 2017.
2. Protocols and Architectures for Wireless Sensor Networks – Holger Karl and Andreas Willig, John Wiley & Sons, 2015.
3. Fundamentals of Wireless Sensor Networks: Theory and Practice – Walteneagus Dargie and Christian Poellabauer, Wiley Series on Wireless Communications and Mobile Computing, 2011.
4. Networking Wireless Sensors – Bhaskar Krishnamachari, Cambridge University Press, 2005.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Wireless Sensor Networks - Practical

**Vertical - 1
Major (Elective) Course - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Wireless Sensor Networks -Practical
Course code: CHMCSV7

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Wireless & Sensor Networks Practical course offers hands-on training in the implementation, simulation, and performance analysis of wireless sensor and ad-hoc networks. Students use tools such as TinyOS, nesC, TOSSIM, and network simulators to study sensor node design, communication protocols, MAC mechanisms, routing strategies, and network behavior in dynamic environments.</p> <p>Relevance:</p> <p>With the increasing adoption of IoT, smart cities, industrial automation, environmental monitoring, and wireless communication technologies, practical knowledge of WSNs and MANETs has become highly relevant. This course bridges the gap between theoretical concepts and practical implementation by enabling students to simulate, evaluate, and understand the behavior of real-world wireless communication systems.</p> <p>Usefulness:</p> <p>The course equips students with practical skills in sensor node programming, event-driven system development, routing protocol implementation, MAC protocol analysis, and energy-efficient communication design. It also enhances their ability to evaluate key wireless network performance metrics, including throughput, latency, packet delivery ratio, and energy consumption.</p> <p>Application:</p> <p>The practical skills acquired through this course are applicable to IoT systems, smart agriculture, healthcare monitoring, industrial automation, environmental sensing, mobile ad-hoc networks, and communication infrastructures. Experience in network simulation, routing analysis, wireless communication, and protocol evaluation supports effective design and deployment of wireless networking solutions.</p>

		<p>Interest:</p> <p>The course is highly engaging as students work with sensor nodes, simulate wireless communication scenarios, analyze routing behavior, evaluate MAC protocols, and study mobile network operations. Hands-on activities involving TinyOS, nesC, TOSSIM, and network simulations provide an interactive learning experience closely aligned with modern wireless communication technologies.</p> <p>Connection with Other Courses:</p> <p>This course complements Wireless Sensor Networks (Theory), Computer Networks, Operating Systems, Embedded Systems, Internet of Things (IoT), Mobile Computing, and Cloud Computing. It reinforces concepts related to protocol implementation, network design, system programming, performance evaluation, and distributed communication systems.</p> <p>Demand in the Industry:</p> <p>Organizations involved in telecommunications, IoT solutions, smart infrastructure, industrial automation, defense systems, and wireless communication networks require professionals with practical expertise in sensor networks and communication protocols. Skills in network simulation, protocol evaluation, and wireless system analysis are increasingly sought after across multiple technology sectors.</p> <p>Job Prospects:</p> <p>Students can pursue career opportunities as Wireless Network Engineers, IoT Developers, Telecom System Analysts, Network Simulation Engineers, Embedded Systems Engineers, Communication Systems Engineers, and Technical Support Specialists. Further specialization can lead to advanced roles in wireless communication research, IoT solution development, and next-generation networking technologies.</p>
2	Vertical	Major (Elective)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours

6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1. To provide practical understanding of wireless sensor node architecture, sensor network components, and TinyOS-based programming concepts.</p> <p>CO(A) 2. To develop skills in simulating wireless sensor networks, ad-hoc networks, and communication scenarios using TOSSIM and network simulation tools.</p> <p>CO(A) 3. To enable implementation, analysis, and evaluation of routing mechanisms, MAC protocols, and wireless communication models in WSN and MANET environments.</p> <p>CO(A) 4. To develop the ability to analyze network performance using metrics such as throughput, packet delivery ratio, delay, connectivity, and communication efficiency.</p> <p>CO(A) 5. To strengthen practical knowledge of wireless sensor networks, mobile communication systems, and network simulation through hands-on experimentation and analysis.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students will be able to:</p> <p>CO1: Identify and analyze sensor node architectures, and develop basic TinyOS-based applications using nesC and TOSSIM.</p> <p>CO2: Simulate wireless sensor networks and ad-hoc networks, and analyze routing behavior and communication mechanisms in dynamic network environments.</p> <p>CO3: Design, implement, and evaluate MAC protocols and wireless communication models for efficient network operation.</p> <p>CO4: Measure and assess network performance using parameters such as throughput, packet delivery ratio, delay, connectivity, and communication efficiency.</p> <p>CO5: Model, simulate, and interpret wireless sensor, MANET, and mobile communication scenarios to evaluate network behavior and performance.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>Unit-I</p> <p>Practical 1: Study of Sensor Node Hardware: To understand the architecture and components of a Wireless Sensor Network (WSN) node, including sensors, sensor motes (nodes), base stations, and graphical user interfaces, and to analyze the functions of the sensing unit, microcontroller, transceiver, memory, and power supply in energy-constrained environments.</p>	

	<p>Practical 2: Exploration of TinyOS Concepts: Examine TinyOS computational concepts such as events, commands, and tasks, and understand the nesC model and components used in sensor network programming.</p> <p>Practical 3: Simulation using TOSSIM: Explore TOSSIM for simulating mote-to-mote radio communication and mote-to-PC serial communication in a sensor network environment.</p> <p>Practical 4: Simulation of Adhoc Network: Create and simulate a simple adhoc network to study dynamic network formation and communication.</p> <p>Practical 5: Routing Table Analysis: Understand, read, and analyze the routing table of a network to study path selection and routing behavior.</p> <p>Unit-II</p> <p>Practical 6: Simulation of MANET: Implement a Mobile Adhoc Network (MANET) simulation to visualize packet animation and packet tracing.</p> <p>Practical 7: Wireless Sensor Network Implementation: Simulate and analyze the operation of a Wireless Sensor Network (WSN).</p> <p>Practical 8: MAC Protocol Simulation: Design and simulate a Medium Access Control (MAC) protocol implementation for efficient communication in a wireless sensor network.</p> <p>Practical 9: Directional Antenna Simulation: Simulate a Mobile Adhoc Network using directional antennas to study their effect on signal strength and network performance.</p> <p>Practical 10: Mobile Network Simulation: Create and simulate a mobile communication network using cell towers, central office servers, web browsers, and web servers, demonstrating connection establishment and data exchange.</p>
--	---

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
	Total 30	

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Wireless Sensor Networks Technology, Protocols, and Applications ,Kazem Sohraby, Daniel Minoli and TaiebZnati, John Wiley & Sons, 2017
2. Protocols and Architectures for Wireless Sensor Network, Holger Kerl, Andreas Willig, John Wiley and Sons, 2015
3. Fundamentals of Wireless Sensor Networks, Theory and Practice, Waltenequs Dargie, Christian Poellabauer , Wiley Series on wireless Communication and Mobile Computing, 2011
4. Networking Wireless Sensors, Bhaskar Krishnamachari , Cambridge University Press, 2005

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

**Vertical – 2
Minor - 2 +2 Credits**

Basics of Mathematics in Real Life – IV : 2 Credits
Practical Based on Basics of Mathematics : 2 Credits
in Real Life – IV

(offered by Mathematics)

**with effect from
Academic Year 2026-2027**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B.Sc.
(Computer Science)
(Mathematics)**

Semester- V

**Title: Basics of Mathematics in Real Life – IV
(offered by Mathematics)**

**Vertical - 2
Minor - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Basics of Mathematics in Real Life – IV

Course Code: CHMMATHV10

Sr. No.	Heading	Particulars
1	Description of the Course:	This course gives introduction to natural numbers, integers, rational numbers, real numbers and complex numbers in detail. Basic concepts like primes and congruences are introduced.
2	Vertical 2	Minor
3	Type	Theory + Practicum (Lectures / Problem Solving / Discussion / Presentation / Case Study / Demonstration etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives:	CO(A) 1. To develop the notions of limits and continuity CO(A) 2. To identify the properties of congruences CO(A) 3. To associate diagrams based on equalities and inequalities of complex numbers CO(A) 4. To recognize various properties of functions and their use in basic counting
8	Course Outcomes:	Student will be able to: CO1: Understand and remember basic concepts of numbers, sets, functions, sequences, and counting principles.

- CO2:** Apply mathematical techniques to solve routine and real-life problems.
- CO3:** Analyze patterns, relationships, and structures in numbers, functions, and sequences.
- CO4:** Justify/check mathematical results using logical reasoning and proof methods.
- CO5:** Construct simple mathematical models, proofs, and examples for real-life situations.

9

Syllabus

UNIT I: Basics of integers, real numbers and complex numbers (15 Hours)

- Natural numbers, Integers, Rational numbers and Irrational numbers
- Introduction to induction in natural numbers via proofs of sums of first n natural numbers and sums of squares and cubes of the first n natural numbers.
- Further applications of induction through problem solving.
- Operations on integers and rational numbers like addition, multiplication and subtraction. Equivalence of two rational numbers.
- Divisibility in integers and basic properties of divisibility
- Definition of prime numbers and statement of fundamental theorem of arithmetic (without proof).
- Greatest common divisor, least common multiple and relation to the product of numbers, Euclid's algorithm (without proof)
- Infinitude of primes (with proof) and existence of irrational numbers (Square-root two is irrational with proof).
- Congruences and their basic properties like solution of linear congruence
- Real number line and properties of real numbers.
- Order on real numbers and relation to the operation on real numbers.
- Definition of a complex number and visualization in the plane. Plotting of complex numbers.
- Operations on complex numbers like addition and multiplication and polar form of complex numbers. DeMoivre's theorem and its proof via induction.
- Plotting of regions in the complex plane defined by equalities and inequalities.

- Definition of a sequence and examples of sequences of natural numbers, integers and real numbers and analyzing the behaviour of sequences pictorially and introduction to the idea of convergence.

UNIT II: Introduction to basic counting and basics of functions (15 Hours)

- Permutations and combinations of distinct objects.
- Examples based on permutations: digits, license plates etc.
- Examples based on combinations: digits, bit strings etc.
- Addition and multiplication principles for counting and illustrations
- Permutations with repetitions (only formula) and examples
- Combinations with repetitions (only formula) and examples
- De Morgan's laws for sets and introduction to functions between sets
- Injective and surjective functions
- Bijective functions, examples and their properties
- Inverse images of sets and their properties
- Limit of a function at a point
- Properties of limits: uniqueness (with proof)
- Computations of limits in various examples
- Definition and examples of continuous functions
- Properties of continuous functions: sums, products and ratios

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 2:00 hour

Format of Question Paper

Q. No.	Structure of the Questions	Marks
1	Attempt ANY THREE (out of five/Six). (Each question of 5 marks, based on unit 1)	15
2	Attempt ANY THREE (out of five/Six). (Each question of 5 marks, based on unit 2)	15

	Total	30
--	--------------	-----------

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is essential component of the evaluation process.

Internal Examination: Continuous Evaluation - 20 marks

Continuous Evaluation through: Quizzes, Class Tests, presentations, projects, role play, creative writing, assignments etc. (at least 3)

Sr. No.	Particulars	Marks
1	A class test of 10 marks is to be conducted during each semester in an Offline mode.	10
2	Project on any one topic related to the syllabus or a quiz (offline/online) on one of the modules.	05
3	Seminar/ group presentation on any one topic related to the syllabus.	05

Paper pattern of the Test (Offline Mode with One hour duration):

Q1: Definitions/Fill in the blanks/ True or False with Justification. (04 Marks: 4 x 1).

Q2: Attempt any 2 from 3 Descriptive questions. (06 marks: 2 × 3)

11

REFERENCES:

1. Burton, D. M/. Elementary Number Theory, McGraw Hill Education
2. Bartle R. G. and Sherbert D. R., Introduction to Real Analysis, John Wiley and Sons.
3. Niven, Ivan, Zuckerman H. S., Montgomery, H. L., An Introduction to the theory of numbers, Wiley, 1972.
4. Richard R. Goldberg, Methods of Real Analysis, John Wiley and Sons.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year
(Mathematics)**

Semester- V

**Title: Practical Based on Basics of Mathematics
in Real Life – IV**

(offered by Mathematics)

**Vertical - 2
Minor - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Practical Based on Basics of Mathematics in Real Life – IV

Course Code: CHMMATHV11

Sr. No.	Heading	Particulars
1	Description of the Course:	This course is based on problem-solving which is a fundamental aspect of any mathematics course. While advanced courses often emphasize the theoretical nature of the subject, engaging in problem-solving reinforces concepts and enhances learners' ability to analyze existing problems and devise solutions. This activity not only motivates learners but also empowers them to formulate new results, propose conjectures, and develop innovative theories.
2	Vertical 2	Minor
3	Type	Practical
4	Credit	2 Credits
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. To develop the notions of limits and continuity CO(A) 2. To identify the properties of congruences CO(A) 3. To associate diagrams based on equalities and inequalities of complex numbers CO(A) 4. To recognize various properties of functions and their use in basic counting	

8	<p>Course Outcomes: Student will be able to:</p> <p>CO1: Apply mathematical techniques to solve routine and real-life problems.</p> <p>CO2: Analyze patterns, relationships, and structures in numbers, functions, and sequences.</p> <p>CO3: Justify/check mathematical results using logical reasoning and proof methods.</p> <p>CO4: Construct simple mathematical models, proofs, and examples for real-life situations.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I: Practical based on basics of integers, real numbers and complex numbers (30 Hours)</p> <ul style="list-style-type: none"> • Integers and divisibility • Computation of greatest common divisor using Euclid’s algorithm • Properties of congruences • Solutions of linear congruences • Plotting of regions in the complex numbers and conversion to polar form • Applications of DeMoivre’s theorem • Sequences and their plotting, Convergence of sequences based on plotting <p>UNIT II: Practical based on Introduction to basic counting and basics of functions (30 Hours)</p> <ul style="list-style-type: none"> • Permutations of distinct objects • Combinations of distinct objects • Permutations and combinations of multisets • Injective, bijective, surjective functions • Inverse images of sets under functions • Limit of a function and examples • Continuity of a function • Arithmetic of continuous functions

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 2:00 hour

Format of Question Paper

Question	Based On	Marks
Q1	Five out of Eight multiple choice questions (four from Unit 1 and four from Unit 2) (LO1 to LO3)	(3 × 5 = 15 Marks)
Q2	Attempt any Two out of Four (Two from Unit 1 and two From Unit 2). (LO 3 and LO 4)	(5 × 2 = 10 Marks)
Q3	Journal	5 (2.5 marks for each Unit 1 & Unit 2)
		Total: 30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is essential component of the evaluation process.

Internal Examination: Continuous Evaluation - 20 marks

Continuous Evaluation through: Quizzes, Class Tests, presentations, projects, role play, creative writing, assignments etc. (at least 3)

	Assessment / Evaluation	Marks
1.	Objective question test	10
2.	Overall performance	5
3.	Viva	5
		Total: 20

	<p>Paper pattern of the Test (Offline Mode):</p> <p>Q1: (Attempt any 5 from 8) Multiple choice questions. (10 marks: 5×2) Duration: 1Hrs</p> <p>While setting question paper four MCQ on unit I and four MCQ on unit II both.</p>
<p>11</p>	<p>REFERENCES:</p> <ol style="list-style-type: none"> 1. Niven H. Zuckerman and H. Montgomery. An Introduction to the Theory of Numbers. John Wiley & Sons. Inc. 2. David M. Burton. An Introduction to the Theory of Numbers. Tata McGraw-Hill Edition. 3. Bartle R. G. and Sherbert D. R., Introduction to Real Analysis, John Wiley and Sons. 4. Richard R. Goldberg, Methods of Real Analysis, John Wiley and Sons. 5. Thomas and Finney, Calculus and Analytical Geometry, Pearson 6. Ajit Kumar and S. Kumaresan, A basic course in real analysis, Chapman and Hall 7. B. V. Limaye and Sudhir Ghorpade, A course in calculus and real analysis, Springer Nature.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Vertical – 4

Vocation Skill Course (VSC) : 2 Credits

**with effect from
Academic Year 2026-2027**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

Title: Ethical Hacking

**Vertical - 4
Vocation Skill Course (VSC): 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Ethical Hacking
Course Code: CHMCSV10

Sr. No.	Heading	Particulars
1.	Description of the course:	<p>Introduction</p> <p>The Ethical Hacking course offers a structured approach to understanding cyber threats, system vulnerabilities, and penetration testing techniques from a defensive perspective. It includes topics such as reconnaissance, scanning, enumeration, privilege escalation, cryptography weaknesses, web application vulnerabilities, malware analysis, wireless security, and the use of exploit frameworks. The course also covers standard penetration testing methodologies while emphasizing legal compliance, ethical responsibilities, and professional conduct in cybersecurity practices.</p> <p>Relevance</p> <p>With the rise in cyber threats targeting governments, organizations, and individuals, ethical hacking has emerged as a vital area within cybersecurity. This course is highly relevant as it provides learners with an understanding of attacker methodologies from a defensive standpoint, enabling them to effectively identify, analyze, and mitigate system vulnerabilities in a proactive manner.</p> <p>Usefulness</p> <p>The course builds practical and analytical skills in vulnerability research, network scanning, web application security assessment, password security analysis, and management of the penetration testing lifecycle. It also enhances defensive cybersecurity thinking by helping learners understand exploit mechanisms, attack surfaces, and effective system hardening strategies.</p> <p>Application</p> <p>Ethical hacking concepts are applied in security auditing, penetration testing, vulnerability assessment, security</p>

operations, web application testing, wireless network evaluation, and compliance verification. The course develops an understanding of OWASP vulnerabilities, SQL injection techniques, buffer overflow concepts, denial-of-service attacks, and exploit frameworks to support secure system design and strengthen organizational cybersecurity.

Interest

The subject is engaging as it examines real-world cyber-attack techniques such as social engineering, session hijacking, ARP poisoning, SQL injection, and malware analysis. By understanding attacker mindsets and exploitation methods, learners gain a technically stimulating experience along with awareness of the broader social impact of cybersecurity threats.

Connection with Other Courses

This course complements Cyber and Information Security, Computer Networks, Operating Systems, Cryptography, Software Testing, and Cloud Security. It integrates foundational networking concepts, encryption principles, system-level knowledge, and secure coding practices into a cohesive, security-focused framework.

Demand in the Industry

There is a growing global demand for ethical hackers, penetration testers, SOC analysts, and cybersecurity consultants. Organizations are actively seeking professionals with expertise in vulnerability assessment, exploit analysis, web application security testing, and secure system configuration to ensure the protection of critical digital infrastructure.

Job Prospects

Students can pursue roles such as penetration testing trainee, cybersecurity analyst, SOC analyst, vulnerability assessment engineer, security consultant, or incident response associate.

		With further specialization and experience, career opportunities expand into advanced roles including red team specialist, security architect, and cybersecurity researcher.
2.	Vertical 4:	VSC
3.	Type:	Practical
4.	Credits:	2 credits (1 credit = 30 Hours of Practical work in a semester)
5.	Hours Allotted:	60 Hours
6.	Marks Allotted:	50 Marks
7.	<p>Course Objectives:</p> <p>CO(A) 1. To introduce fundamental concepts, terminology, and legal and ethical frameworks relevant to ethical hacking.</p> <p>CO(A) 2. To develop understanding of reconnaissance, scanning, enumeration, and system exploitation techniques from a security defense perspective.</p> <p>CO(A) 3. To impart knowledge of vulnerabilities in web applications, networks, and wireless systems along with their mitigation techniques.</p> <p>CO(A) 4. To familiarize students with exploit tools, frameworks, and structured penetration testing methodologies.</p> <p>CO(A) 5. To develop ethical awareness and professional skills in cybersecurity reporting and documentation practices.</p>	
8.	<p>Course Outcomes: After successful completion of this course, students would be able to –</p> <p>CO1: Understand and analyze cyber-attack methodologies such as reconnaissance, scanning, and privilege escalation techniques.</p> <p>CO2: Identify and evaluate vulnerabilities in web applications, networks, wireless systems, and authentication mechanisms.</p> <p>CO3: Assess cryptographic weaknesses, session management flaws, and malware behavior from a defensive cybersecurity perspective.</p> <p>CO4: Conduct structured penetration testing and produce professional vulnerability assessment and remediation reports.</p> <p>CO5: Propose effective security controls, mitigation techniques, and system hardening practices in accordance with established legal and ethical guidelines.</p>	
9.	<p>Syllabus</p> <p>Unit-I</p>	

Practical 1: Ethical Reconnaissance, Information Gathering, and Digital Footprint Analysis

- a) Write a practical to perform reconnaissance using Google Dorking, search engines, OSINT techniques, and Whois lookup to gather publicly available information about a target domain. Use a web browser and publicly available resources to document findings, collect screenshots as evidence, and prepare a digital footprint analysis report.

Practical 2: Network Discovery and Port Scanning

- a) Write a practical to perform network reconnaissance using networking commands such as ipconfig/ifconfig, ping, netstat, and traceroute, and use Nmap to conduct ACK, SYN, FIN, NULL, and XMAS port scans to identify active hosts, open ports, and running services. Record scan results and analyze discovered services.

Practical 3: Vulnerability Assessment and Network Attack Analysis

- a) Write a practical to identify security weaknesses in a controlled network environment and demonstrate Address Resolution Protocol (ARP) poisoning concepts for understanding man-in-the-middle attack techniques and associated risks. Analyze network behavior and document security observations.

Practical 4: Password Security, Encryption, and Auditing

- a) Write a practical to evaluate password security through hash analysis and dictionary-based password recovery techniques on authorized test credentials. Analyze password strength, compare hash values, and document audit findings.

Practical 5: Network Traffic Capture and Protocol Analysis

- a) Write a practical to capture, monitor, and analyze network traffic using Wireshark, examine protocol behavior, identify communication patterns, and document security-relevant observations. Save packet captures and perform protocol-level analysis.

Unit-II

Practical 6: Web Application Security Testing – Cross-Site Scripting (XSS)

- a) Write a practical to perform security testing on a locally hosted vulnerable web application, identify and demonstrate persistent Cross-Site Scripting (XSS) vulnerabilities in a controlled environment, and document mitigation techniques. Analyze browser behavior and captured evidence related to the attack.

Practical 7: Session Management and Authentication Security Testing

- a) Write a practical to analyze session management mechanisms and demonstrate session impersonation concepts using browser-based tools to understand

session security vulnerabilities and countermeasures. Examine cookies and session tokens generated during testing.

Practical 8: Denial-of-Service Attack Simulation and Detection

- a) Write a practical to simulate a Denial-of-Service (DoS) attack against a locally hosted service in a controlled laboratory environment, analyze the impact on network resources and services, identify attack signatures, and document mitigation strategies.

Practical 9: Web Application Security Testing – SQL Injection

- a) Write a practical to perform SQL Injection testing on a locally hosted vulnerable web application using SQLMap. Identify vulnerable input fields, analyze database responses, document discovered vulnerabilities, and recommend appropriate mitigation techniques.

Practical 10: Malware Behavior Analysis Using Python

- a) Write a practical to develop and analyze a simple Python-based keylogger in a controlled laboratory environment to understand malware behavior, event monitoring, and associated security risks. Analyze generated logs, observe system activity, and document security implications and preventive measures.

10.

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 02:00 hours

Format of Question Paper

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11.	REFERENCES: <ol style="list-style-type: none">1. CEH official Certified Ethical Hacking Review Guide, Wiley India Edition2. Certified Ethical Hacker: Michael Gregg, Pearson Education3. Certified Ethical Hacker: Matt Walker, TMH.4. Bill Nelson, Amelia Philips and Christopher Steuart, "Guide to computer forensics and investigations", course technology, 6th edition.5. Kevin Mandia, Chris Prosise, "Incident Response and computer forensics", Tata McGrawHill

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- V

**Title: Community Engagement
Project (CEP)**

**Vertical - 6
Continuing Education Program (CEP): 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Community Engagement Project (CEP)
Course Code: CHMCSV11

Name of Faculty: - Science & Technology

Name of Programme: - B.Sc. (Computer Science)

Duration: - 30 hrs (Field Work+ Survey) + 15hrs (Discussion + Report Writing): Total - 45 hrs

Objectives:

The Community Engagement Project should:

- Encourage **real-world interactions**: students must engage with real communities, organizations, or stakeholders.
- Include **design of survey tools**: like questionnaires, interviews, focus group discussions.
- Involve **on-site visits or virtual interactions** for data collection, feedback, or implementation.
- Require **use of computer science knowledge**: such as databases, web/app prototyping, data analysis, or visualization.
- Promote **social innovation and digital inclusion**.
- Preferably include **group projects** (up to 4–5 students), with structured faculty guidance.
- Encourage **field-based problem-solving** and **prototype development** of digital tools/websites/apps that address community issues.

Indicative Topics for CEP

Sr. No.	Project Title	Description
1.	Digital Literacy for Senior Citizens	Conduct awareness and hands-on sessions for elderly individuals on smartphone usage, online payments, and safe browsing.
2.	AI Awareness in Schools	Develop simple, engaging activities to introduce school students to the basics of Artificial Intelligence.
3.	E-Governance and Citizen Portals	Field study on public usability of platforms like UMANG, Digilocker; suggest UI/UX improvements.
4.	Promoting Online Safety Among Teenagers	Design a campaign around privacy settings, cyberbullying, and digital detox techniques.
5.	Gender-Sensitive App Design for Public Safety	Survey women's safety concerns and build prototype apps with SOS, alerts, and location sharing.
6.	Accessibility Audit of Government Websites	Evaluate local and national websites for accessibility by differently-abled users and recommend enhancements.

7.	Gamified App for Sustainability	Create a prototype app encouraging eco-friendly choices using reward-based gameplay.
8.	Vernacular IT Helpdesk Model	Study local communities and design low-cost support kiosks in local languages.
9.	Empowering Women Entrepreneurs Digitally	Assess digital challenges faced by small-scale women-led businesses and offer tech-based solutions.
10.	Cybersecurity for Gig Workers	Engage delivery partners or cab drivers in interactive sessions on OTP scams, fake links, and password hygiene.
11.	Digital Empowerment of Street Vendors	Study their tech use and design simple tools or guides for payment apps and inventory.
12.	QR Code-Based Learning Trails in Parks	Build educational QR trails in gardens to teach kids general science, logic, or coding trivia.
13.	Impact of Screen Time on Children	Survey parents and teachers to analyze digital addiction and offer balanced screen usage tips.
14.	Digital Footprint Audit for Students	Build a checklist/tool for students to assess and clean their online presence.
15.	NGO Website/App Makeover	Choose a grassroots NGO and build them a working digital presence (web/app prototype).
16.	AI Chatbot for Community Queries	Prototype a chatbot for common local questions (garbage pickup, bus timings, helpline numbers).
17.	Smart Plugs: Awareness and Field Testing	Introduce local homes to energy-saving IoT plugs and monitor adoption/benefits.
18.	Vernacular Mobile Diary for Farmers	Co-design a mobile diary for farmers to record weather, crop schedules, and sales in native languages.
19.	Gamified App for Traffic Rules	Make a quiz app for delivery agents and school students to learn road signs and safe practices.
20.	Awareness on Facial Recognition & Consent	Educate the public about surveillance risks, biometric privacy, and consent in public spaces.
21.	Cybersecurity Comic Book	Design a story-based comic for children in English / Hindi / Marathi to raise awareness about safe internet usage.
22.	Data Privacy Sessions for Self Help Groups	Conduct community workshops on password safety, fake apps, and sharing of personal information.
23.	Feedback Portal for Municipal Services	Develop and test a mobile/web portal for local residents to report issues to municipal bodies.
24.	Digital Identity for Local Artists	Help street or local artists set up digital portfolios, QR payments, and social media presence
25.	Assistive Technology for Visually Impaired	Organize awareness and demo sessions on screen readers, voice assistants, and OCR scanners.

Note: The topics are indicative and the faculty members should allot Field Projects that are relevant and important as per core Subject. The Field Project may be taken individual or in a group up to 5 students with proper guidance from Faculty.

Deliverables for Field Project:

- **Field Visit Report** – Documenting observations, photographs, and key findings from on site interactions.
- **Project Planning and Design Document** – Detailing survey instruments, interview questions, data collection plans, and any supporting materials such as app/website prototypes, workflow diagrams, or training module outlines.
- **Final Report** – Covering the problem statement, literature review, methodology, data collection and analysis, conclusions, and recommendations.
- **Presentation for Viva** – Summarizing project objectives, methodology, outcomes, and key learnings for oral examination.

EVALUATION PATTERN

Evaluation during CEP Program involves two key components:

1. External Evaluation 60%
2. Internal Evaluation 40%

EVALUATION CHART

I. Internal Evaluation by Guide – 20 Marks

Criteria	Marks
Attendance, Community interactions completion and interaction with Supervisor	10
Overall Report quality	10
Total	20

II. External Examination – 30 Marks

Criteria	Marks
Objectives, Literature Review, Methodology, Data Analysis, Conclusion and Recommendations	15
Overall Project Report Structure and Style	05
Presentation Skills & Communication	10
Total	30

* Please see the Guidelines for Community Engagement Project for UG Students, as per NEP 2020



HSNC Board's
Smt. Chandibai Himathmal Mansukhani College, Ulhasnagar
(Autonomous)
Affiliated to the University of Mumbai

Bachelor of Science
(Computer Science)
(Self-Financing Course)

Semester – VI

Choice Based and Credit Based syllabus
as per NEP 2020 with effect from the
Academic Year 2026-2027

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Vertical – 1

Data Science	: 2 Credits
Cloud Computing	: 2 Credits
Software Project Management	: 2 Credits
Computer Science Practical 6	: 2 Credits
Mini Project – II	: 2 Credits
Information Retrieval	: 2 Credits
Information Retrieval Practical	: 2 Credits

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Data Science

**Vertical - 1
Major (Mandatory) Course - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Data Science
Course Code: CHMCSV11

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Data Science course provides a comprehensive introduction to the end-to-end process of extracting meaningful insights from data. It encompasses data collection, preprocessing, exploratory data analysis, statistical inference, machine learning algorithms, model evaluation, visualization, data management, and ethical considerations. The course develops a structured understanding of how data-driven decision systems are designed, implemented, and applied to real-world problems.</p> <p>Relevance:</p> <p>Data Science plays a vital role in modern decision-making across industries such as finance, healthcare, retail, governance, and technology. The course is highly relevant as it integrates statistical methods, machine learning techniques, data visualization, and data governance practices to prepare students for analytics-driven environments and data-centric problem solving.</p> <p>Usefulness:</p> <p>The course develops practical competence in data cleaning, feature engineering, predictive model construction, performance evaluation, cross-validation, and hyperparameter optimization. It also enhances analytical thinking, statistical interpretation, and the capability to derive actionable insights from raw datasets for informed decision-making.</p> <p>Application:</p> <p>Data Science concepts are extensively used in applications such as customer segmentation, fraud detection, predictive maintenance, recommendation systems, demand forecasting, sentiment analysis, and risk assessment. Core techniques including regression, classification, clustering, ensemble methods, and ROC analysis form the foundation of modern enterprise analytics and intelligent decision-making systems.</p> <p>Interest:</p>

		<p>The subject is engaging as it helps students convert raw data into meaningful insights through systematic analysis and modeling. Practical exposure to visualization methods, hypothesis testing, feature engineering, and comparative model evaluation enhances both logical reasoning and creative problem-solving skills, making the learning experience interactive and application-oriented.</p> <p>Connection with Other Courses:</p> <p>This course complements subjects such as Artificial Intelligence, Database Management Systems, Statistics, Machine Learning, Cloud Computing, and Data Visualization by integrating programming, statistical analysis, and computational thinking within a comprehensive data analytics framework.</p> <p>Demand in the Industry:</p> <p>The growing adoption of data-driven technologies has created significant demand for professionals with expertise in data analytics, predictive modeling, and machine learning. Organizations across diverse sectors seek individuals skilled in data preprocessing, model evaluation, data visualization, ETL processes, and data governance to derive actionable insights and support informed strategic and operational decision-making.</p> <p>Job Prospects:</p> <p>Students completing this course can pursue careers as Data Analysts, Business Intelligence Analysts, Junior Data Scientists, Machine Learning Associates, or Analytics Consultant Trainees. With further specialization and professional experience, they may advance to roles such as Data Scientists, Machine Learning Engineers, Data Engineers, Analytics Consultants, or Analytics Leads.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours

6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To introduce the fundamental concepts and scope of Data Science and its interrelation with Artificial Intelligence, Machine Learning, and Business Intelligence.</p> <p>CO(A) 2.To build proficiency in data preprocessing, feature engineering, and exploratory data analysis techniques.</p> <p>CO(A) 3.To develop understanding of supervised and unsupervised learning algorithms along with basic optimization methods.</p> <p>CO(A) 4.To equip students with skills in model evaluation, validation techniques, and hyperparameter optimization.</p> <p>CO(A) 5.To promote awareness of ethical, privacy, governance considerations in responsible data handling, along with an introduction to automated machine learning (AutoML) approaches.</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1:Apply data preprocessing, transformation, feature selection, and statistical methods to conduct effective exploratory data analysis.</p> <p>CO2:Develop and evaluate machine learning models for classification, regression, and clustering tasks using suitable performance metrics.</p> <p>CO3:Apply cross-validation, hyperparameter tuning, and automated machine learning (AutoML) approaches for optimal model selection and performance enhancement.</p> <p>CO4:Design and interpret effective visualizations to communicate analytical insights clearly and effectively.</p> <p>CO5:Demonstrate understanding of data governance, privacy regulations (including GDPR and DPDP Act 2023), and ethical data handling practices.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Data Science: Definition and scope of Data Science, Applications and domains of Data Science, Comparison with other fields like Business Intelligence (BI), Artificial Intelligence (AI), Machine Learning (ML), and Data Warehousing/Data Mining (DW-DM), Different types of data: structured, unstructured, semi-structured; Data sources: databases, files, APIs, web scraping, sensors, social media • Data Preprocessing: Data cleaning: handling missing values, outliers, duplicates; Data transformation: scaling, normalization, encoding categorical variables; Feature selection: selecting relevant features/columns; Data merging: combining multiple datasets; Data wrangling techniques: reshaping, pivoting, aggregating • Exploratory Data Analysis (EDA): Feature engineering: creating new features, 	

handling time-series data; Dummification: converting categorical variables into binary indicators; Feature scaling: standardization, normalization; Data visualization techniques: histograms, scatter plots, box plots, etc.; Descriptive statistics: mean, median, mode, standard deviation, etc.; Hypothesis testing: t-tests, chi square tests, ANOVA, etc.

- **Machine Learning Algorithms:** Supervised learning: classification and regression, Unsupervised learning: clustering and dimensionality reduction, Bias-variance tradeoff, underfitting, and overfitting, Simple linear regression, Multiple linear regression, Stepwise regression, Logistic regression for classification, Decision Trees and Random Forests, Support Vector Machines (SVM), Artificial Neural Networks (ANN), Ensemble Learning: Boosting and Bagging, K-Nearest Neighbors (K-NN), Gradient Descent for optimization.

UNIT II:

- **Model Evaluation, Selection and Metrics:** Techniques for evaluating, model performance: accuracy, precision, recall, F1-score, Confusion, matrix and ROC – AUC analysis, evaluating models for imbalanced, datasets, Cross-validation: k-fold cross-validation, stratified cross- validation, Hyperparameter tuning and model selection
- **Data Visualization and Communication:** Principles of effective data visualization, Types of visualizations: bar charts, line charts, scatter plots, etc. Data storytelling: communicating insights through visualizations
- **Tools and Libraries:** Introduction to popular libraries and technologies, used in Data Science like Pandas, NumPy, Sci-kit Learn, etc. Visualization tools: matplotlib, seaborn, Tableau, etc.
- **Data Management:** Introduction to data management activities, Data pipelines: data extraction, transformation, and loading (ETL), Data governance and data quality assurance, Data privacy and security considerations
- **Data Ethics and Privacy:** Data lifecycle ethics (collection, use, reuse), Privacy risks and consent, Basic overview of GDPR and DPDP Act 2023, Responsible data handling practices.
- **AutoML:** Introduction to AutoML, need and significance of AutoML in Data Science, AutoML pipeline, difference between AutoML and traditional Machine Learning, AutoML tools such as PyCaret and H2O AutoML, applications of AutoML in real-world domains.
- **AI-Assisted Data Analysis:** Introduction, natural language querying, automated visualization, insight generation, limitations, and applications.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Data Science from Scratch First Principles with Python- Joel Grus O'reilly, 2nd Edition
2. Advancing into Analytics From Excel to Python and R, George Mount, Oreilly, First Edition
3. Introduction to Machine Learning with Python, Andreas C. Muller, Sarah Guido, Oreilly, First Edition
4. Doing Data Science, Rachel Schutt and Cathy O'Neil, O'Reilly, 2013
5. Mastering Machine Learning with R, Cory Lesmeister, PACKT Publication, 2015
6. Hands-On Programming with R, Garrett Golemund, 1st Edition, 2014
7. An Introduction to Statistical Learning, James, G., Witten, D., Hastie, T., Tibshirani, R., Springer, 2015

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Cloud Computing

Vertical - 1

Major (Mandatory) Course - 2 Credits

**with effect from
Academic Year 2026-2027**

Title: Cloud Computing
Course Code: CHMCSV12

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Cloud Computing course provides students with a foundation in distributed computing concepts, web services (SOAP and REST), virtualization technologies, and contemporary cloud architectures. It also explores core cloud characteristics, service and deployment models, and fundamental security principles. The course includes hands-on experience through simulation using CloudSim and practical exposure to cloud platforms such as OpenStack and AWS.</p> <p>Relevance:</p> <p>Cloud computing is a core technology that underpins modern digital infrastructure, enterprise applications, and scalable services. This course is highly relevant as organizations are increasingly shifting their workloads to cloud platforms and need professionals skilled in virtualization, cloud security, orchestration, and infrastructure management.</p> <p>Usefulness:</p> <p>The course equips students with knowledge of web service protocols such as WSDL, SOAP, and REST, along with virtualization using KVM, cloud deployment models, and secure cloud software practices. It also develops the ability to design, simulate, and manage cloud environments while understanding key concepts such as scalability, elasticity, and resource optimization.</p> <p>Application:</p> <p>Cloud computing technologies play a vital role in modern IT infrastructures, supporting Software-as-a-Service (SaaS) applications, enterprise systems, DevOps practices, scalable web services, virtual private cloud (VPC) architectures, private cloud deployments, and secure multi-tenant environments. Platforms such as CloudSim, OpenStack, and AWS are extensively used</p>

		<p>for cloud infrastructure management, virtualization, resource provisioning, and the deployment of cloud-native applications.</p> <p>Interest:</p> <p>The course is engaging as it explores how large-scale systems are designed, deployed, and managed dynamically in distributed environments. It provides hands-on experience with virtualization tools, OpenStack deployment, AWS architecture design, and cloud orchestration, making the subject highly practical, immersive, and aligned with real-world enterprise systems.</p> <p>Connection with Other Courses:</p> <p>This course complements subjects such as Computer Networks, Operating Systems, Cyber Security, Software Engineering, Distributed Systems, and DevOps. It integrates concepts of virtualization, resource management, secure software development, and scalable architecture design.</p> <p>Demand in the Industry:</p> <p>Cloud computing skills are highly sought after across industries such as IT services, fintech, healthcare, e-commerce, and enterprise technology. Expertise in AWS architecture, private cloud deployment, virtualization, cloud security, infrastructure orchestration, and resource management is increasingly valued for designing, deploying, and maintaining scalable and secure cloud-based solutions.</p> <p>Job Prospects:</p> <p>Students can explore roles such as Cloud Engineer Trainee, DevOps Associate, Cloud Administrator, AWS Associate, Virtualization Engineer, or Infrastructure Support Engineer. With advanced experience, career progression may lead to roles such as Cloud Architect, Site Reliability Engineer (SRE), or Cloud Security Specialist, along with other advanced positions in cloud infrastructure and platform engineering.</p>
2	Vertical 1	Major (Mandatory)

3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. To introduce the fundamental principles of distributed systems, web services, and virtualization technologies. CO(A) 2. To build an understanding of cloud computing models, characteristics, and deployment architectures. CO(A) 3. To impart knowledge of cloud security fundamentals and secure software development practices in cloud environments. CO(A) 4. To familiarize students with cloud simulation tools and private cloud platforms such as OpenStack. CO(A) 5. To develop an understanding of AWS-based cloud infrastructure and architectural design solutions.	
8	Course Outcomes: After successful completion of this course, students would be able to - CO1: To understand and distinguish between cloud service models, deployment models, and virtualization techniques. CO2: To design and implement web services using SOAP and REST principles within distributed environments. CO3: To configure, deploy, and analyze virtualized and cloud environments using tools such as KVM and OpenStack. CO4: To assess cloud security requirements and implement secure development practices along with policy enforcement. CO5: To demonstrate knowledge of AWS architecture and design scalable and efficient cloud-based solutions.	
9	<p style="text-align: center;">Syllabus</p> UNIT I: <ul style="list-style-type: none"> • Cloud Computing Basics: Web Services – Distributed Computing, Parallel Computing, WSDL structure, SOAP- Structure of SOAP Message (In JAX-WS), SOAP Messaging Architecture, SOAP Header, Client-side SOAP Handler, REST- What is REST? HTTP methods, Java API for RESTful Web Services (JAX-RS) • Virtualization: Characteristics of Virtualized Environments, Pros and Cons of Virtualization. Virtualization using KVM, Creating virtual machines, oVirt - management tool for virtualization environment. 	

- **Introduction to Cloud Computing:** Definition, Types of Clouds, Deployment of software solutions and web applications, Types of Cloud Platforms, Essential characteristics – On-demand self-service, Broad network access, Location independent resource pooling ,Rapid elasticity , Measured service, Comparing cloud providers with traditional IT service providers

UNIT II:

- **Cloud Computing Software Security fundamentals:** Cloud Information Security Objectives, Confidentiality, Integrity, Availability, Cloud Security Services, Relevant Cloud Security Design Principles, Secure Cloud Software Requirements, Secure Development practices, Approaches to Cloud Software Requirement Engineering, Cloud Security Policy Implementation.
- **CloudSim:** Introduction to Simulator, understanding CloudSim simulator, CloudSim Architecture(User code, CloudSim, GridSim, SimJava) Understanding Working platform for CloudSim,
- **OpenStack:** Introduction to OpenStack, OpenStack test-drive, Basic OpenStack operations, OpenStack CLI and APIs, Tenant model operations, Quotas, Private cloud building blocks, Controller deployment, Networking deployment, Block Storage deployment, Compute deployment, deploying and utilizing OpenStack in production environments, Building a production environment, Application orchestration using OpenStack Heat
- **AWS:** Architecting on AWS, Building complex solutions with Amazon Virtual Private Cloud (Amazon VPC)

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Enterprise Cloud Computing Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
2. Mastering Cloud Computing, Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, Tata McGraw Hill Education Private Limited, 2013
3. OpenStack in Action, V. K. CODY BUMGARDNER, Manning Publications Co, 2016
4. Java Web Services Up and Running 2nd edition, Martin Kalin, O'Reilly (2013)
5. Pro Power Shell for Amazon Web Services, Brian Beach, Apress, 2014
6. OpenStack Essentials, Dan Radez, PACKT Publishing, 2015
7. OpenStack Operations Guide, Tom Fifield, Diane Fleming, Anne Gentle, Lorin Hochstein, Jonathan Proulx, Everett Toews, and Joe Topjian, O'Reilly Media, Inc., 2014
8. <https://www.openstack.org>

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Software Project Management

Vertical - 1

Major (Mandatory) Course - 2 Credits

**with effect from
Academic Year 2026-2027**

Title: Software Project Management
Course Code: CHMCSVI3

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Software Project Management course introduces students to structured methodologies and industry best practices for planning, executing, monitoring, and closing software projects. It covers key areas such as project initiation, scope definition, effort estimation, scheduling, cost management, quality assurance, risk management, resource allocation, agile methodologies, governance, and ethical considerations in software-driven environments.</p> <p>Relevance:</p> <p>In the digital era, effective software delivery requires not only strong technical expertise but also sound project planning and management practices. This course is highly relevant as modern organizations seek professionals who can efficiently manage project scope, schedule, cost, quality, and risks while adopting agile methodologies and collaborating in distributed development environments.</p> <p>Usefulness:</p> <p>The course provides students with practical knowledge of Work Breakdown Structures (WBS), project scheduling techniques, Earned Value Management (EVM), risk assessment, procurement planning, stakeholder management, and agile frameworks. It also develops managerial, analytical, and decision-making skills required to effectively lead and manage software projects.</p> <p>Application:</p> <p>Software project management principles are applied across IT services, product development, startup ecosystems, enterprise digital transformation initiatives, and cloud-based solution deployments. Key techniques such as schedule tracking, cost control, vendor management, agile sprint planning, and stakeholder communication are widely practiced in the industry.</p>

		<p>Interest:</p> <p>This subject uniquely blends technical planning with leadership, negotiation, and strategic decision-making. It further explores agile management, virtual team coordination, Six Sigma principles, governance frameworks, and ethical responsibilities, providing practical insights into real-world technology project management.</p> <p>Connection with Other Courses:</p> <p>This course aligns with subjects such as Software Engineering, Cloud Computing, DevOps, Quality Assurance, and Enterprise Systems. It bridges technical development processes with managerial frameworks, enhancing students' comprehensive understanding of software lifecycle management.</p> <p>Demand in the Industry:</p> <p>Demand for professionals skilled in project coordination, agile methodologies, cost management, and stakeholder engagement continues to grow. IT firms, consulting companies, fintech organizations, and startups increasingly rely on trained project managers and team leads to ensure timely and cost-effective software delivery.</p> <p>Job Prospects:</p> <p>Students can explore career opportunities as Project Coordinator, Associate Project Manager, Scrum Master Trainee, Business Analyst, or Operations Executive. With experience and advanced skills, they can progress to roles such as Project Manager, Program Manager, Agile Coach, or IT Governance Specialist.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks

7	<p>Course Objectives:</p> <p>CO(A) 1.To introduce the fundamental processes and knowledge areas of project management in software development.</p> <p>CO(A) 2.To develop competencies in scope definition, scheduling, cost estimation, and quality management.</p> <p>CO(A) 3.To build capability in risk assessment, procurement management, and stakeholder coordination.</p> <p>CO(A) 4.To provide an understanding of agile methodologies and modern digital project management tools.</p> <p>CO(A) 5.To foster leadership skills, ethical responsibility, and governance practices in project environments.</p>
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: To effectively plan and manage software projects using scope, time, cost, and quality management techniques.</p> <p>CO2: To utilize risk assessment methods, earned value analysis, and procurement strategies for effective project control.</p> <p>CO3: To apply agile methodologies and manage iterative software development processes.</p> <p>CO4: To demonstrate leadership abilities, stakeholder communication, and conflict resolution skills within project teams.</p> <p>CO5: To follow governance frameworks and uphold ethical standards in professional project management practice.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Project Management: Definition and characteristics of a project, Importance of project management, Project management processes and knowledge areas • Project Selection, Initiation and scope Management: Project selection criteria and methods, Project initiation and charter development, Stakeholder identification and analysis, Scope planning and definition, Work Breakdown Structure (WBS) development, Scope verification and change control • Project Time & Cost Management: Activity definition and sequencing, Estimating activity durations and resources, Developing the project schedule, Schedule control and monitoring, Cost estimation techniques, Budget development and monitoring, Earned Value Management (EVM), Cost control and analysis • Project Quality & Risk Management: Quality planning and standards, Quality

assurance and control, Process improvement and Six Sigma concept, Risk Management-Risk identification and assessment, Risk response planning, Risk monitoring and control

UNIT II:

- **Project Resource & Procurement Management:** Team development and roles, Staffing, training, and motivation, Conflict resolution and communication management, Procurement planning and contract types, Solicitation, source selection, and contract administration, Vendor management and relationship building
- **Project Integration Management:** Project integration processes and activities, Change management and project closure, Lessons learned and knowledge transfer
- **Agile Project & Management in the Digital Age:** Agile principles and methodologies, Agile project planning and execution, Managing iterative and incremental development, Role of technology in project management, Virtual teams and distributed project management, Tools and software for project planning and collaboration
- **Effective People Management in Projects:** Leadership styles and characteristics, Team building and motivation techniques, Emotional intelligence in project management, Stakeholder identification and analysis, Stakeholder engagement and communication strategies, Conflict resolution and negotiation skills
- **Project Governance and Ethics:** Project governance structures and accountability, Ethical considerations in project management, Professional responsibility and codes of conduct

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Project Management for Business and Technology, 3rd edition, Pearson Education. John M. Nicholas, 2000
2. Information Technology Project Management, by Jack T. Marchewka, 4th Wiley India 2013.
3. A Guide to the Project Management Body of Knowledge (PMBOK® Guide) Sixth Edition 6th Edition, Project Management Institute, 2017
4. Introduction to Software Project Management by Adolfo Villafiorita · 2016, CRC press, e book format.
5. Project Management Professional Workbook, Claudia M. Baca, Patti M. Jansen, Sybex Publication, 2013
6. Project Management, by S. J. Mantel, J. R. Meredith and etal., 1st edition, Wiley India, 2009.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Computer Science Practical 6

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Computer Science Practical 6
Course Code: CHMCSV14

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>Computer Science Practical – 6 provides integrated hands-on training in Data Science and Cloud Computing technologies. It enables students to carry out data analysis, statistical modeling, visualization, and machine learning experiments, along with practical experience in developing and deploying web services, virtualization environments, and cloud-based solutions using industry-standard platforms.</p> <p>Relevance:</p> <p>This course is highly relevant as it integrates two key technological domains driving digital transformation—Data Science and Cloud Computing. Its practical focus ensures that students not only understand analytical and cloud concepts but also gain hands-on experience in implementing and deploying real-world solutions using modern tools and frameworks.</p> <p>Usefulness:</p> <p>The course provides students with applied skills in data preprocessing, hypothesis testing, regression modeling, clustering, principal component analysis (PCA), and data storytelling through visualization tools. In addition, it builds competencies in web service design (SOAP/REST), API integration, virtualization using KVM, cloud deployment using platforms such as OpenStack and AWS, and service orchestration, thereby strengthening overall technical versatility.</p> <p>Application:</p> <p>Data Science practicals are applied in predictive analytics, classification modeling, clustering analysis, regression forecasting, and decision support systems. Cloud Computing practicals focus on distributed service deployment, API consumption, virtual infrastructure provisioning, REST/SOAP service design, and enterprise cloud environment configuration.</p>

Interest:

The course is engaging as it enables students to progress from analyzing structured datasets and performing hypothesis testing to deploying cross-platform web services and configuring cloud infrastructures. Hands-on activities such as Google Maps API integration, OpenStack network setup, AWS workflow execution, and MTOM-based multimedia transfer provide an immersive, industry-aligned learning experience.

Connection with Other Courses:

This course aligns with Data Science, Artificial Intelligence, Cloud Computing, Database Management Systems, Software Engineering, and Computer Networks. It integrates analytical modeling with scalable infrastructure deployment, thereby strengthening interdisciplinary competencies across the curriculum.

Demand in the Industry:

The industry shows strong demand for professionals proficient in data analytics, machine learning modeling, API development, virtualization, and cloud infrastructure management. Hands-on experience with tools such as Pandas, Scikit-Learn, AWS, OpenStack, KVM, and RESTful services significantly improves employability in areas like data engineering, cloud engineering, and DevOps.

Job Prospects:

Students completing this course can pursue roles such as Data Analyst Trainee, Junior Data Scientist, Cloud Engineer Associate, DevOps Trainee, API Developer, or Infrastructure Support Engineer. The combination of data analytics and cloud deployment skills builds a strong foundation for advanced specialization in areas such as Artificial Intelligence, Data Engineering, and Cloud Architecture.

2	Vertical 1	Major (Mandatory)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. To build hands-on competence in preparing data, applying statistical techniques, and implementing machine learning models. CO(A) 2. To gain practical experience in designing and building web services using SOAP and REST architectures. CO(A) 3. To develop operational skills in configuring virtualization and managing cloud infrastructure using KVM, OpenStack, and AWS. CO(A) 4. To enhance the ability to analyze and present insights using data visualization, storytelling techniques, and model evaluation methods. CO(A) 5. To apply data-driven modeling approaches for designing and deploying scalable cloud-based solutions.	
8	Course Outcomes: - After successful completion of this course, students would be able to - CO1: Apply techniques such as data preprocessing, hypothesis testing, regression, clustering, and PCA for analyzing real-world datasets. CO2: Design and implement SOAP and REST web services and incorporate external APIs into application development. CO3: Set up, configure, and manage virtual machines and cloud environments using tools such as KVM, OpenStack, and AWS. CO4: Interpret and communicate analytical outcomes through visualization, structured data storytelling, and AI-assisted exploratory data analysis using natural language queries and generated insights. CO5: Combine data science methods with cloud deployment strategies for building and managing scalable solutions.	
9	Syllabus UNIT I: Practical 1: Introduction to Excel <ul style="list-style-type: none"> • Perform conditional formatting on a dataset using various criteria. • Create a pivot table to analyze and summarize data. 	

	<ul style="list-style-type: none"> • Use VLOOKUP function to retrieve information from a different worksheet or table. • Perform what-if analysis using Goal Seek to determine input values for desired output. <p>Practical 2: Data Frames and Basic Data Pre-processing</p> <ul style="list-style-type: none"> • Read data from CSV and JSON files into a data frame. • Perform basic data pre-processing tasks such as handling missing values and outliers. • Manipulate and transform data using functions like filtering, sorting, and grouping. <p>Practical 3: Feature Scaling and Dummification</p> <ul style="list-style-type: none"> • Apply feature-scaling techniques like standardization and normalization to numerical features. • Perform feature dummification to convert categorical variables into numerical representations. <p>Practical 4: Hypothesis Testing</p> <ul style="list-style-type: none"> • Formulate null and alternative hypotheses for a given problem. • Conduct a hypothesis test using appropriate statistical tests (e.g., t-test, chi square test). • Interpret the results and draw conclusions based on the test outcomes. <p>Practical 5: ANOVA (Analysis of Variance)</p> <ul style="list-style-type: none"> • Perform one-way ANOVA to compare means across multiple groups. • Conduct post-hoc tests to identify significant differences between group means. <p>Practical 6: Regression and Its Types</p> <ul style="list-style-type: none"> • Implement simple linear regression using a dataset. • Explore and interpret the regression model coefficients and goodness-of-fit measures. • Extend the analysis to multiple linear regression and assess the impact of additional predictors. <p>Practical 7: Logistic Regression and Random Forest</p> <ul style="list-style-type: none"> • Build a logistic regression model to predict a binary outcome.
--	---

- Evaluate the logistic regression model's performance using classification metrics such as accuracy, precision, recall, F1-score, and confusion matrix.
- Construct a Random Forest classifier for the same dataset and compare its performance with the logistic regression model using appropriate evaluation metrics.

Practical 8: K-Means Clustering

- Apply the K-Means algorithm to group similar data points into clusters.
- Determine the optimal number of clusters using elbow method or silhouette analysis.
- Visualize the clustering results and analyze the cluster characteristics.

Practical 9: Principal Component Analysis (PCA)

- Perform PCA on a dataset to reduce dimensionality.
- Evaluate the explained variance and select the appropriate number of principal components.
- Visualize the data in the reduced-dimensional space

Practical 10: Data Visualization and Storytelling

- Create meaningful visualizations using data visualization tools
- Combine multiple visualizations to tell a compelling data story.
- Present the findings and insights in a clear and concise manner.

Practical 11: Demonstration of AI-Assisted Data Analysis Tool

- Upload a dataset into an AI-assisted data analysis platform, such as Julius AI.
- Perform exploratory data analysis using natural language queries and generate visualizations.
- Analyze AI-generated insights, summaries, and basic predictions, and interpret the results.

Unit-II

Practical 1: Web Service + Cross-Platform Client Calls

Cross-Platform Currency Conversion Service (INR to USD) with Java and .NET Clients: Design and deploy a simple web service that accepts an amount in Indian Rupees (INR) and returns the

	<p>equivalent value in US Dollars (USD) using a configurable exchange rate. Implement two client applications—one in Java and one in .NET—that invoke the service, pass parameters, receive the response, and display the converted value. Include input validation, basic error handling, and logs of requests/responses.</p> <p>Practical 2: SOAP Web Service Development and Deployment of a Simple SOAP-Based Web Service: Create a SOAP web service exposing at least two operations (e.g., convertCurrency(), getServiceStatus()), publish the WSDL, and test it using a SOAP client (SoapUI or programmatic client). Demonstrate request/response XML structure, handling invalid parameters, and service endpoint accessibility.</p> <p>Practical 3: REST Web Service Design and Implementation of a Simple RESTful Service with CRUD APIs: Develop a REST service that exposes APIs using HTTP methods (GET/POST/PUT/DELETE) for a small resource (e.g., Student, Course, or Task). Ensure responses are in JSON, implement basic validations, HTTP status codes, and test using Postman/cURL. Document endpoints and sample payloads.</p> <p>Practical 4: Consuming Google Search / Google Maps REST API: Integration of External REST API: Google Maps-Based Place Search and Geolocation Viewer: Build a client application that consumes Google Maps REST APIs (e.g., Geocoding / Places) to accept a user query (place/address) and display extracted results such as latitude, longitude, formatted address, and optionally show it on a map UI. Handle API key configuration, HTTP request building, JSON parsing, rate-limit/error responses, and present output in a user-friendly format</p> <p>Practical 5: KVM Virtualization Setup Installation and Configuration of KVM for Virtual Machine Provisioning: Install and configure KVM virtualization on a Linux host system. Verify hardware virtualization support, configure required packages (KVM, QEMU, libvirt, virt-manager), and create at least one VM with defined CPU/RAM/disk. Demonstrate VM lifecycle operations: start/stop/snapshot, and document networking mode (NAT/bridge) used.</p> <p>Practical 6: Image/Video Upload-Download Using MTOM Binary Data Transfer using MTOM: Upload and Download of Multimedia Files: Implement a SOAP-based file transfer service</p>
--	---

	<p>using MTOM (Message Transmission Optimization Mechanism) to upload and download image/video files between client and server. Ensure efficient binary transfer (not base64-heavy payloads), validate file size/type, store files server-side, and verify successful retrieval with integrity (file opens correctly).</p> <p>Practical 7: FOSS-Cloud VSI Functionality for IaaS + Storage Implementing FOSS-Cloud IaaS: Virtual Server Infrastructure (VSI) with Storage Provisioning: Create a mini IaaS-style setup using FOSS cloud tools where users can provision a virtual server instance with selected CPU/RAM/disk and attach storage (block/object as supported by the chosen tool). Demonstrate instance creation, storage creation, attachment/mounting, and basic monitoring/usage reporting.</p> <p>Practical 8: FOSS-Cloud VSI Functionality for PaaS Implementing FOSS-Cloud PaaS: Deploying a Web Application on a Platform Service: Use an open-source PaaS (or PaaS-like layer) to deploy a sample application (e.g., simple web app/API). The platform should manage runtime, deployment, scaling configuration (manual/auto if possible), and service bindings (database/cache optional). Demonstrate deploy, update (version change), view logs, and service health.</p> <p>Practical 9: AWS Flow Framework Workflow Application AWS Flow Framework: Hello-World Workflow with Activity and Worker Programs: Develop a workflow-based application using AWS Flow Framework where a workflow triggers an activity that prints “Hello World” to the console. Implement workflow and activity interfaces (contracts), their implementations, workflow coordination logic, and worker programs to host and execute them. Demonstrate workflow execution and tracking of workflow/activity invocation.</p> <p>Practical 10: OpenStack Setup: User Network + Private Network Creation OpenStack Networking Practical: Creating Tenant User Network and Private Network Topology: Set up (or use an existing) OpenStack environment and create a tenant configuration including user/tenant, private network, subnet, and optionally a router connected to an external network. Launch an instance in the private network, assign IP addressing, validate connectivity (ping/SSH if applicable), and document the network topology created.</p>
10	<p align="center">Scheme of Examination and Assessment Pattern</p> <p align="center">Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 02:00 hours</p> <p align="center">Format of Question Paper</p>

Question	Practical Questions Based on	Marks
Q.1.	Unit – I	15
Q.2.	Unit – II	15
		Total 30

Note:

- **Certified Journal is compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process.

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15
2.	Journal	05
		Total 20

11

REFERENCES:

1. Hands-On Programming with R, Garrett Golemund, 1st Edition, 2014
2. Mastering Machine Learning with R, Cory Lesmeister, PACKT Publication, 2015
3. Java Web Services Up and Running 2nd edition, Martin Kalin, O'Reilly (2013)
4. OpenStack Operations Guide, Tom Fifield, Diane Fleming, Anne Gentle, Lorin Hochstein, Jonathan Proulx, Everett Toews, and Joe Topjian, O'Reilly Media, Inc., 2014
5. Introduction to Machine Learning with Python, Andreas C. Muller, Sarah Guido, Oreilly, First Edition
6. Mastering Cloud Computing, Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, Tata McGraw Hill Education Private Limited, 2013
7. OpenStack in Action, V. K. CODY BUMGARDNER, Manning Publications Co, 2016

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Mini Project – II

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Mini Project – II (Research-Oriented Project with Implementation)
Course Code: CHMCSVI5

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>Mini Project – II is a research-focused course designed to develop students’ analytical, experimental, and scholarly skills through systematic investigation and implementation. The course guides students from research problem identification and literature survey to methodology design, experimentation, result analysis, and research paper writing in IEEE format.</p> <p>Relevance:</p> <p>In the era of rapid technological advancement, the ability to conduct structured research and validate solutions experimentally is highly relevant. This course prepares students for higher studies, research roles, and innovation driven industry environments by emphasizing scientific inquiry and evidence-based evaluation.</p> <p>Usefulness:</p> <p>The course equips students with competencies in literature review, research gap identification, dataset selection, experimental design, model implementation, performance evaluation, and academic writing. It strengthens critical thinking, comparative analysis, hypothesis validation, and documentation skills required in research and R&D domains.</p> <p>Application:</p> <p>Research-oriented implementation can be applied in areas such as Artificial Intelligence, Data Science, Cyber Security, Cloud Computing, IoT, or Software Engineering. Students design prototypes, evaluate models against baselines, and generate measurable results, mirroring real world research and innovation workflows.</p> <p>Interest:</p> <p>The course is intellectually stimulating as students explore emerging technologies, analyze prior research, design experimental frameworks, and validate findings through implementation. The opportunity to publish or present work in different formats enhances academic motivation and professional confidence.</p>

		<p>Connection with Other Courses:</p> <p>Mini Project – II integrates knowledge from Artificial Intelligence, Data Science, Cyber Security, Cloud Computing, Software Engineering, and Statistics. It builds upon earlier theoretical and practical courses, transforming technical knowledge into research-driven innovation and structured experimentation.</p> <p>Demand in the Industry:</p> <p>Industries engaged in AI, analytics, cybersecurity, cloud solutions, and advanced software development value professionals who can conduct experimentation, benchmark solutions, and interpret performance metrics. Research oriented skills are especially valuable in R&D divisions, product innovation teams, and technology consulting firms.</p> <p>Job Prospects:</p> <p>Completion of this course prepares students for roles such as research assistant, data scientist trainee, AI/ML engineer, cybersecurity analyst, or innovation associate. It also provides a strong foundation for postgraduate studies, research fellowships, and publication-oriented academic careers.</p>
2	Vertical 1	Major (Mandatory)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To develop skills in research problem identification and structured literature review.</p> <p>CO(A) 2.To enable design of experimental methodology and selection of appropriate models and tools.</p> <p>CO(A) 3.To provide hands-on experience in implementing and evaluating research prototypes.</p> <p>CO(A) 4.To cultivate analytical skills for result interpretation and hypothesis validation.</p> <p>CO(A) 5.To train students in professional research documentation and IEEE-format paper writing.</p>	

<p>8</p>	<p>Course Outcomes: After successful completion of this course, students would be able to -</p> <p>CO1: Formulate and justify a research problem supported by literature review and gap analysis CO2: Design and implement an experimental framework including dataset selection, model development, and evaluation metrics. CO3: Conduct and analyze experiments using appropriate performance comparisons and baseline studies. CO4: Interpret results critically, identify limitations, and propose future research directions. CO5: Prepare and present a structured research paper and working prototype following IEEE standards.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <p style="text-align: center;">Research Planning, Literature Survey & Methodology Design Phase</p> <p>Research Problem Identification: Selection of topic, Problem statement formulation, Research objectives, Hypothesis (if applicable) Literature Review: Study of minimum 5–8 research papers, Identification of research gap, Comparative analysis, Summary of existing techniques Research Methodology Design: Dataset selection or data collection, Algorithm/model selection, Tool selection, Experimental setup planning, Definition of evaluation metrics Research Design Documentation: Research Proposal, Literature Review Matrix, System Model / Architecture Diagram, Experimental Design Plan</p> <p>UNIT II:</p> <p style="text-align: center;">Implementation, Experimentation & Evaluation Phase</p> <p>System Implementation: Model building, Algorithm implementation, Experimental setup, Code optimization Experimental Evaluation: Model training/testing, Performance metric calculation, Baseline comparison Result Analysis: Graphical representation, Interpretation of results, Validation of hypothesis, Identification of limitations Research Paper Writing: Abstract, Introduction, Related Work, Methodology, Results, Conclusion & Future Scope</p> <p>Final Deliverables</p> <ul style="list-style-type: none"> • Working prototype • Research Paper (IEEE format, 6–8 pages minimum) • GitHub Repository • Presentation

10	Evaluation for Mini Project		
	Paper – 50 Marks		
	Mini Project – II		
	Internal Evaluation (To be assessed by the Project Guide)	Component	Marks
		Research Problem Identification	5
		Literature Review	5
		Methodology & Implementation Progress	5
		Internal Seminar / Review	5
	External Evaluation (To be assessed by External Examiner)	Component	Marks
		Prototype / Experimental Implementation	10
Result Analysis & Interpretation		10	
Research Report / Paper		5	
Viva Voce		5	

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Information Retrieval

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Information Retrieval

Course Code: CHMCSV16

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Information Retrieval (IR) course provides students with a comprehensive understanding of techniques used to organize, index, search, and retrieve relevant information from large-scale document repositories. The course covers fundamental concepts such as document representation, inverted indexing, retrieval models, query processing, spelling correction, text classification, clustering, web search architectures, link analysis, learning-to-rank methods, and emerging topics including summarization and question answering systems</p> <p>Relevance:</p> <p>Information Retrieval is a core technology behind modern search engines, digital libraries, e-commerce portals, recommendation platforms, and enterprise information systems. As the volume of digital content continues to expand rapidly, efficient retrieval, ranking, and personalization mechanisms have become essential for delivering accurate and user-oriented information access, making this subject highly relevant in today's data-driven environment.</p> <p>Usefulness:</p> <p>The course enables students to develop a strong foundation in indexing structures, TF-IDF weighting schemes, cosine similarity measures, probabilistic retrieval techniques, evaluation methodologies, and ranking strategies. It enhances analytical and problem-solving abilities required to design efficient retrieval systems, assess their effectiveness, and optimize search quality in large information repositories.</p> <p>Application:</p> <p>Concepts of Information Retrieval are widely applied in web search engines, scholarly databases, content management systems, legal and medical search platforms, recommendation engines, and multilingual information services. Techniques such as PageRank, HITS, clustering, text categorization, and learning-to-rank algorithms play a significant role in improving information discovery and user experience across diverse applications.</p>

		<p>Interest:</p> <p>The course offers an engaging exploration of how search systems process and rank massive collections of documents within fractions of a second. Topics including link analysis, duplicate and near-duplicate detection, ranking methodologies, cross-lingual retrieval, user behavior analysis, and A/B testing provide practical insights into the functioning of real-world search and recommendation systems.</p> <p>Connection with Other Courses:</p> <p>This course closely integrates with Data Structures and Algorithms through indexing and efficiency considerations, Artificial Intelligence through ranking and decision-making models, Data Science through analytics and evaluation metrics, Machine Learning through classification and ranking techniques, and Natural Language Processing through text representation, retrieval, and summarization tasks.</p> <p>Demand in the Industry:</p> <p>Organizations across search technology, e-commerce, finance, social media, digital publishing, and content management sectors actively seek professionals with expertise in retrieval systems, ranking optimization, recommendation technologies, and multilingual information processing. The growing dependence on intelligent search solutions continues to increase industry demand for such skills.</p> <p>Job Prospects:</p> <p>Students completing this course can pursue careers as Search Engineers, Information Retrieval Analysts, Data Analysts, NLP Engineer Trainees, Information Systems Analysts, or Recommendation System Developers. With advanced specialization, they may also explore opportunities in search engine optimization analytics, machine learning for ranking systems, and research-oriented roles in information access and retrieval technologies</p>
2	Vertical 1	Major (Elective)
3	Type	Theory + Practicum + Teaching methods (Lectures, Problem Solving, Discussion, Presentation, Case Study, Simulations, Interdisciplinary Approach, etc)

4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To provide a strong foundation in the principles, architecture, and operational challenges of Information Retrieval systems.</p> <p>CO(A) 2.To develop an understanding of document representation, indexing structures, retrieval models, and document ranking methodologies.</p> <p>CO(A) 3.To enable students to analyze and evaluate the effectiveness of Information Retrieval systems using established performance measures and experimental techniques.</p> <p>CO(A) 4.To impart knowledge of web search mechanisms, link analysis algorithms, and modern learning-to-rank frameworks used in search applications.</p> <p>CO(A) 5.To acquaint students with emerging areas of Information Retrieval, including text summarization, recommender systems, question answering, and multilingual information access</p>	
8	<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Develop and implement document indexing and retrieval solutions using Boolean, Vector Space, and Probabilistic Retrieval models.</p> <p>CO2: Assess the performance of Information Retrieval systems using standard evaluation measures such as Precision, Recall, F-Measure, and ranking-based metrics.</p> <p>CO3: Apply link analysis algorithms, clustering methods, text classification techniques, and learning-to-rank approaches to enhance retrieval effectiveness.</p> <p>CO4: Examine web crawling methodologies and near-duplicate detection techniques employed in large-scale search and retrieval environments.</p> <p>CO5: Demonstrate understanding of advanced Information Retrieval applications, including text summarization, question answering systems, recommender systems, and multilingual information retrieval.</p>	
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I:</p> <ul style="list-style-type: none"> • Introduction to Information Retrieval (IR) Systems: Definition and objectives of Information Retrieval, Components and architecture of an IR system, Challenges, applications, and modern search environments, Overview of search engines, digital libraries, and recommender systems. • Document Indexing, Representation, and Compression: Inverted index construction and compression techniques, Document pre-processing and representation, Term 	

weighting schemes: TF-IDF and BM25, Storage and retrieval of indexed documents.

- **Retrieval Models:** Boolean retrieval model: Boolean operators and query processing, Vector Space Model: TF-IDF, cosine similarity, query-document matching, Probabilistic retrieval models, Introduction to semantic retrieval and dense vector representations.
- **Query Processing and Spelling Correction:** Challenges of spelling errors in queries and documents, Edit distance and string similarity measures, Techniques for spelling correction in Information Retrieval systems.
- **Performance Evaluation:** Evaluation metrics: Precision, Recall, F-measure, Average Precision, Mean Average Precision (MAP), Normalized Discounted Cumulative Gain (NDCG), Test collections and relevance judgments, Experimental design, significance testing.
- **Text Categorization and Filtering:** Text classification algorithms: Naïve Bayes, Support Vector Machines, and Transformer-based classifiers, Feature selection and dimensionality reduction, Applications of text categorization, filtering.
- **Text Clustering for Information Retrieval:** Clustering techniques: K-means and hierarchical clustering, Evaluation of clustering results, Clustering for query expansion, result grouping.

UNIT-II:

- **Web Information Retrieval:** Modern web search architecture and challenges, Crawling and indexing web pages, Link analysis and PageRank algorithm, Semantic search and vector-based retrieval, Knowledge graphs in web search.
- **Link Analysis and its Role in IR Systems:** Web graph representation and link analysis algorithms, HITS and PageRank algorithms, Applications of link analysis in IR systems.
- **Crawling and Near-Duplicate Page Detection:** Web page crawling techniques: breadth-first, depth-first, focused crawling, Near-duplicate page detection algorithms, Handling dynamic web content during crawling.
- **Learning to Rank:** Algorithms and techniques, Supervised learning for ranking: RankSVM, RankBoost, Pairwise and Listwise learning-to-rank approaches, Transformer-based ranking using BERT and Sentence-BERT, Evaluation metrics for learning to rank.
- **Advanced Topics in IR:** Text Summarization: extractive, abstractive and neural summarization methods; Question Answering: open-domain and generative QA approaches; Recommender Systems: collaborative filtering, content-based filtering.
- **AI-Driven Information Retrieval:** Large Language Models (LLMs) for search, Dense retrieval using embeddings, Retrieval-Augmented Generation (RAG), Vector databases and similarity search, Hallucination mitigation and grounding techniques in retrieval

systems.

- **Cross-Lingual and Multilingual Retrieval:** Challenges and techniques for cross-lingual retrieval, Machine translation for IR, Multilingual document representations and query translation, Retrieval for low-resource and Indian languages, Evaluation Techniques for IR Systems.
- **User-based evaluation:** user studies and surveys, Test collections and benchmarking, Online evaluation methods: A/B testing and interleaving experiments, Explainability and trustworthiness metrics for AI-powered IR systems.

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 01:00 hours

Format of Question Paper

Q. No.	Structure of the Questions	Marks
Q.1.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 1)	15
Q.2.	Attempt ANY THREE out of SIX (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation – 20 marks

	Assessment / Evaluation	Marks
1.	Class Test – It should be conducted using any learning management system such as MOODLE. The test should have 10 MCQs which should be solved in a time duration of 20 minutes .	10
2.	Presentation, Demonstration, Case Study, Seminar, Posters, Mini-Project, Role play etc.	10
		Total 20

11

REFERENCES:

1. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, —Modern Information Retrieval: The Concepts and Technology behind Search, Second Edition, ACM Press Books
2. C. Manning, P. Raghavan, and H. Schütze, —Introduction to Information Retrieval, Cambridge University Press
3. Ricci, F, Rokach, L. Shapira, B. Kantor, —Recommender Systems HandbookI, First Edition.
4. Bruce Croft, Donald Metzler, and Trevor Strohman, Search Engines: Information Retrieval in Practice, Pearson Education.
5. Stefan Buttcher, Charlie Clarke, Gordon Cormack, Information Retrieval: Implementing and Evaluating Search Engines, MIT Press.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: Information Retrieval Practical

**Vertical - 1
Major (Mandatory) - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Information Retrieval -Practical

Course Code: CHMCSVI7

Sr. No.	Heading	Particulars
1	Description of the Course:	<p>Introduction:</p> <p>The Information Retrieval Practical course focuses on the hands-on implementation of fundamental and advanced IR techniques using benchmark datasets such as the Cranfield Collection, 20 Newsgroups, BBC News Dataset, SNAP Web Graphs, and LETOR. The course covers text preprocessing, indexing structures, retrieval models, evaluation methodologies, document classification, clustering, web crawling, link analysis, and learning-to-rank techniques through practical experimentation.</p> <p>Relevance:</p> <p>As search engines, recommendation platforms, and large-scale information systems continue to drive modern digital ecosystems, practical knowledge of Information Retrieval has become increasingly important. This course bridges theoretical concepts and real-world applications by enabling students to design, implement, and evaluate retrieval systems using standard datasets and performance measures.</p> <p>Usefulness:</p> <p>The course develops practical skills in text processing, inverted index construction, TF-IDF weighting, cosine similarity computation, Boolean retrieval, retrieval evaluation using precision-recall measures, MAP calculation, document classification using Naïve Bayes and SVM, clustering analysis, and ranking evaluation through NDCG. It enhances analytical thinking, experimental design, and quantitative evaluation capabilities.</p> <p>Application:</p> <p>Practical Information Retrieval techniques are widely</p>

used in web search engines, enterprise search solutions, digital libraries, news aggregation systems, recommendation platforms, and multilingual information access applications. Competencies in indexing, ranking, clustering, and link analysis are directly applicable to search, analytics, and data-driven technology domains.

Interest:

The course offers an engaging learning experience by enabling students to build search systems, compare retrieval approaches, implement spelling correction mechanisms, visualize evaluation metrics, analyze web graphs using PageRank, and develop learning-to-rank models. The use of real-world datasets provides meaningful insights into practical retrieval challenges and solutions.

Connection with Other Courses:

This course complements Information Retrieval (Theory), Data Structures and Algorithms, Artificial Intelligence, Machine Learning, Natural Language Processing, and Data Science. It integrates concepts from text analytics, statistical learning, graph processing, and performance evaluation into a comprehensive practical framework.

Demand in the Industry:

There is a growing demand for professionals with expertise in search technologies, ranking systems, recommendation engines, document analytics, and information access solutions. Organizations in search, e-commerce, finance, digital content, and analytics sectors actively seek individuals skilled in indexing, retrieval modeling, ranking optimization, and evaluation methodologies.

Job Prospects:

		Students completing this course can pursue roles such as Search Engineer, NLP Engineer Trainee, Data Analyst, Machine Learning Associate, Recommendation Systems Developer, and Information Systems Analyst. Advanced expertise in Information Retrieval can further lead to careers in search architecture, ranking optimization, intelligent information systems, and research-oriented positions in information access technologies.
2	Vertical 1	Major (Elective)
3	Type	Practical
4	Credit	2 credits (1 credit = 30 Hours of Practical work in a semester)
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	<p>Course Objectives:</p> <p>CO(A) 1.To develop practical competency in text preprocessing, indexing structures, and the implementation of information retrieval models.</p> <p>CO(A) 2.To enable students to evaluate retrieval systems using standard performance measures such as Precision, Recall, MAP, and NDCG.</p> <p>CO(A) 3.To provide hands-on exposure to text classification, clustering techniques, and link analysis methods.</p> <p>CO(A) 4.To facilitate the implementation of web crawling strategies and ranking algorithms used in search systems.</p> <p>CO(A) 5.To strengthen experimental evaluation skills through comparative analysis using benchmark information retrieval datasets.</p>	
8	<p>Course Outcomes: -</p> <p>After successful completion of this course, students would be able to -</p> <p>CO1: Design, implement, and evaluate inverted indexes, Boolean retrieval systems, vector space models, and TF-IDF-based ranking approaches.</p> <p>CO2: Calculate and interpret information retrieval evaluation metrics including Precision, Recall, F1-Score, Average Precision (AP), Mean Average Precision (MAP), and NDCG.</p> <p>CO3: Apply text classification techniques such as Naïve Bayes and SVM, and perform clustering using K-Means and Hierarchical methods on textual datasets.</p> <p>CO4: Develop web crawling solutions and implement link analysis algorithms such as PageRank on graph-based datasets.</p> <p>CO5: Construct, evaluate, and compare learning-to-rank models against conventional retrieval baselines using standard benchmark datasets.</p>	
9	Syllabus	

Unit-I

Practical 1: Text Preprocessing on Cranfield Collection

Perform tokenization, stopword removal, stemming, vocabulary analysis, and term frequency computation on the Cranfield dataset.

- Understand basic text preprocessing techniques.
- Prepare raw text data for Information Retrieval applications.
- Analyze the impact of preprocessing on vocabulary size.

Practical 2: Inverted and Positional Index Construction

Build inverted and positional indexes for efficient term and phrase-based document retrieval.

- Construct efficient indexing structures for document collections.
- Enable phrase query processing using positional information.
- Improve retrieval speed and accuracy.

Practical 3: Boolean Retrieval System

Implement Boolean retrieval using Cranfield queries and analyze retrieval results using relevance judgments.

- Apply Boolean operators for document retrieval.
- Execute simple and complex search queries.
- Compare retrieved results with relevance judgments.

Practical 4: TF-IDF and Vector Space Model

Compute TF-IDF weights and perform ranked retrieval using the Vector Space Model.

- Calculate term importance using TF-IDF.
- Represent documents and queries as vectors.
- Rank documents based on similarity scores.

Practical 5: Retrieval Evaluation Metrics

Evaluate retrieval effectiveness using Precision, Recall, F1-Score, Precision@K, and MAP.

- Measure the effectiveness of retrieval systems.
- Compare retrieval performance using standard metrics.
- Analyze the quality of search results.

Unit-II

Practical 6: Web Crawling and Indexing of Web Pages

Implement web page crawling and indexing.

- Understand the process of web page crawling and document collection.
- Create indexes for efficient retrieval of crawled web documents.
- Perform search and retrieval on indexed web pages.

Practical 7: Link Analysis using HITS Algorithm

Implement link analysis on a web graph using the HITS algorithm.

- Understand web graph representation and link structures.
- Compute hub and authority scores for web pages.
- Analyze the importance of web pages based on link relationships.

Practical 8: PageRank Algorithm

Implement the PageRank algorithm for ranking web pages.

- Understand the concept of PageRank and web page importance.
- Compute PageRank scores using a web graph.
- Analyze the effect of link structures on page ranking.

Practical 9: Near-Duplicate Page Detection

Detect near-duplicate web pages using similarity measures.

- Understand techniques for identifying duplicate and near-duplicate content.
- Apply similarity detection algorithms on web pages.
- Analyze content redundancy within document collections.

Practical 10: Learning to Rank Techniques

Implement learning-to-rank approaches for Information Retrieval.

- Understand ranking methodologies used in modern search systems.
- Apply pairwise and listwise learning-to-rank algorithms such as RankSVM and RankBoost.
- Compare and evaluate ranking performance using different approaches.

Practical 11: Text Summarization and Question Answering

Implement advanced Information Retrieval applications for summarization and question answering.

- Understand extractive and abstractive text summarization techniques.
- Retrieve precise answers for user queries using question-answering methods.

	<ul style="list-style-type: none"> Analyze the effectiveness of summarization and question-answering systems. <p>Practical 12: Recommender Systems and Cross-Lingual Retrieval Implement recommender systems and multilingual retrieval techniques.</p> <ul style="list-style-type: none"> Understand collaborative and content-based recommendation approaches. Perform retrieval across multiple languages using query translation techniques. Analyze retrieval effectiveness in multilingual environments. <p>Practical 13: Evaluation of Information Retrieval Systems Evaluate Information Retrieval systems.</p> <ul style="list-style-type: none"> Calculate evaluation metrics for learning to rank. Perform user-based evaluation and benchmarking. Conduct online evaluation using A/B testing and interleaving experiments. 																								
10	<p style="text-align: center;">Scheme of Examination and Assessment Pattern Paper – 50 Marks</p> <p>External Examination: Semester End External - 30 marks Time: 02:00 hours Format of Question Paper</p> <table border="1" data-bbox="555 1031 1268 1182"> <thead> <tr> <th>Question</th> <th>Practical Questions Based on</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td>Q.1.</td> <td>Unit – I</td> <td>15</td> </tr> <tr> <td>Q.2.</td> <td>Unit – II</td> <td>15</td> </tr> <tr> <td colspan="2"></td> <td style="text-align: right;">Total 30</td> </tr> </tbody> </table> <p>Note:</p> <ul style="list-style-type: none"> Certified Journal is compulsory for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination. Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is an essential component of the evaluation process. <p>Internal Examination: Continuous Evaluation – 20 marks</p> <table border="1" data-bbox="414 1541 1442 1799"> <thead> <tr> <th></th> <th>Assessment / Evaluation</th> <th>Marks</th> </tr> </thead> <tbody> <tr> <td>1.</td> <td>Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.</td> <td>15</td> </tr> <tr> <td>2.</td> <td>Journal</td> <td>05</td> </tr> <tr> <td colspan="2"></td> <td style="text-align: right;">Total 20</td> </tr> </tbody> </table>	Question	Practical Questions Based on	Marks	Q.1.	Unit – I	15	Q.2.	Unit – II	15			Total 30		Assessment / Evaluation	Marks	1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15	2.	Journal	05			Total 20
Question	Practical Questions Based on	Marks																							
Q.1.	Unit – I	15																							
Q.2.	Unit – II	15																							
		Total 30																							
	Assessment / Evaluation	Marks																							
1.	Practical Assignments / Experiments / Hands-On Tests / Presentations / Demonstrations / Online Class Test / Case Studies / Mini-Projects / Complete the code block / Develop the incomplete portion of the code etc.	15																							
2.	Journal	05																							
		Total 20																							
11	<p>REFERENCES:</p> <ol style="list-style-type: none"> Ricardo Baeza-Yates and Berthier Ribeiro-Neto, —Modern Information 																								

	<p>Retrieval: The Concepts and Technology behind Search, Second Edition, ACM Press Books</p> <ol style="list-style-type: none">2. C. Manning, P. Raghavan, and H. Schütze, —Introduction to Information Retrieval, Cambridge University Press3. Ricci, F, Rokach, L. Shapira, B. Kantor, —Recommender Systems Handbook, First Edition.4. Bruce Croft, Donald Metzler, and Trevor Strohman, Search Engines: Information Retrieval in Practice, Pearson Education.5. Stefan Buttcher, Charlie Clarke, Gordon Cormack, Information Retrieval: Implementing and Evaluating Search Engines, MIT Press.
--	--

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

**Vertical – 2
Minor - 2 +2 Credits**

Basics of Mathematics in Real Life – IV : 2 Credits
Practical Based on Basics of Mathematics : 2 Credits
in Real Life – IV

(offered by Mathematics)

**with effect from
Academic Year 2026-2027**

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year
(Mathematics)**

Semester- VI

Title: Basics of Mathematics in Real Life – V

(offered by Mathematics)

Vertical - 2

Minor - 2 Credits

**with effect from
Academic Year 2026-2027**

Title: Basics of Mathematics in Real Life – V

Course Code: CHMMATHVI12

Sr. No.	Heading	Particulars
1	Description of the Course:	This course gives a gentle introduction to one variable calculus, which is very helpful to students across various disciplines. Important concepts like convergence and recurrence relations are introduced. The linearization process that plays a vital role all across mathematics is introduced through the definition of differentiability. Important theorems based on this concept are presented. A quick flavour of integration, a key concept of mathematics, with extremely important applications is also given.
2	Vertical 2	Minor
3	Type & Teaching Method	Theory + Practicum (Lectures / Problem Solving / Discussion / Presentation / Case Study / Demonstration etc.)
4	Credit	2 Credits
5	Hours allotted	30 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. To develop the notion of recurrence CO(A) 2. To identify sequences that are convergent CO(A) 3. To associate a Taylor series expansion to differentiable functions CO(A) 4. To recognize the importance of continuity and differentiability in locating zeroes of functions.	

<p>8</p>	<p>Course Outcomes: Student will be able to:</p> <p>CO1: Understand and remember definitions and examples of sequences, convergence, differentiability, derivatives, and integration.</p> <p>CO2: Apply rules of sequences, recurrence relations, differentiation, and integration to solve standard and real-life problems.</p> <p>CO3: Analyze the behavior of sequences and functions, including convergence, differentiability, and extrema.OC4: to outline methods to find zeroes of functions</p> <p>CO4: Justify/check results using logical reasoning, limit arguments, and standard theorems of calculus.</p> <p>CO5: Construct recurrence relations, Taylor expansions, and mathematical models for real-life applications.</p>
<p>9</p>	<p style="text-align: center;">Syllabus</p> <p>UNIT I</p> <ul style="list-style-type: none"> • Sequences of real numbers and examples • Convergence of a sequence of real numbers: definition • Examples of convergent sequences • Examples of non-convergent (divergent) sequences • Sums and products of sequences (convergent sequences) • Sums and products of sequences (not necessarily convergent) • Sandwich theorem for sequences and applications • Sequences defined by recurrence relation: definition and examples • Statement for existence of solution of homogeneous recurrence relation (without proof) • Examples of solving homogeneous recurrence relations: linear and quadratic • Examples of solving homogeneous recurrence relations: cubic and quartic • Examples of non-homogeneous recurrence relations and statement for existence of solution. • Non homogeneous recurrence relation examples (root does not match with root of the associated characteristic polynomial) • Non homogeneous recurrence relation examples (root matches with a root of the associated characteristic polynomial) • Setting up recurrence relations in simple examples

UNIT II

- Definition of differentiability of a function
- Examples of differentiable and non-differentiable functions
- Calculation of the derivative using first principles: simple examples
- Sums, products of differentiable functions are differentiable (with proofs)
- Ratios of differentiable functions are differentiable (with proof)
- Applications of the above rules in various examples and intermediate value property
- Examples of functions which are differentiable finitely many times and those which are differentiable infinitely many times.
- Rolle's theorem (only statement) and applications
- Cauchy and Lagrange's mean value theorems and applications (only statements)
- Derivative of the inverse of a function and examples
- Taylor expansion of a function at a point: definition and remainder term
- Examples for computation of Taylor series of functions
- Applications of derivatives in real life problems (finding maxima, minima)
- Definition of (Riemann) integration and examples
- Basic properties of Riemann integration and easy examples

10

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 1:00 hour

Format of Question Paper

Q. No.	Structure of the Questions	Marks
1	Attempt ANY THREE (out of five/Six). (Each question of 5 marks, based on unit 1)	15
2	Attempt ANY THREE (out of five/Six). (Each question of 5 marks, based on unit 2)	15
	Total	30

Internal Examination: Continuous Evaluation - 20 marks

Continuous Evaluation through: Quizzes, Class Tests, presentations, projects, role play, creative writing, assignments etc. (at least 3)

Sr.No.	Particulars	Marks
1	A class test of 10 marks is to be conducted during each semester in an Offline mode.	10
2	Project on any one topic related to the syllabus or a quiz (offline/online) on one of the modules.	05
3	Seminar/ group presentation on any one topic related to the syllabus.	05

Paper pattern of the Test (Offline Mode with One hour duration):

Q1: Definitions/Fill in the blanks/ True or False with Justification. (04 Marks: 4 x 1).

Q2: Attempt any 2 from 3 Descriptive questions. (06 marks: 2 × 3)

11 REFERENCES:

1. Bartle R. G. and Sherbert D. R., Introduction to Real Analysis, John Wiley and Sons.
2. Kenneth Rosen, Discrete Mathematics and its applications, McGraw Hill.
3. Richard R. Goldberg, Methods of Real Analysis, John Wiley and Sons.
4. Brualdi, Richard A., Introductory Combinatorics, Pearson Education.

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year
(Mathematics)**

Semester- VI

**Title: Practical Based on Basics of Mathematics
in Real Life – V**

(offered by Mathematics)

**Vertical - 2
Minor - 2 Credits**

**with effect from
Academic Year 2026-2027**

Title: Practical Based on Basics of Mathematics in Real Life – V

Course Code: CHMMATHVI13

Sr. No.	Heading	Particulars
1	Description of the Course:	This course is based on problem-solving which is a fundamental aspect of any Mathematics course. While advanced courses often emphasize the theoretical nature of the subject, engaging in problem-solving reinforces concepts and enhances learners' ability to analyze existing problems and devise solutions. This activity not only motivates learners but also empowers them to formulate new results, propose conjectures, and develop innovative theories.
2	Vertical 1	Minor
3	Type	Practical
4	Credit	2 Credits
5	Hours allotted	60 Hours
6	Marks allotted	50 Marks
7	Course Objectives: CO(A) 1. To develop the notion of recurrence CO(A) 2. To identify sequences that are convergent CO(A) 3. To associate a Taylor series expansion to differentiable functions CO(A) 4. To recognize the importance of continuity and differentiability in locating zeroes of functions	
8	Course Outcomes: Student will be able to:	

	<p>CO1: Apply rules of sequences, recurrence relations, differentiation, and integration to solve standard and real-life problems.</p> <p>CO2: Analyze the behavior of sequences and functions, including convergence, differentiability, and extrema. OC4: to outline methods to find zeroes of functions</p> <p>CO3: Justify/check results using logical reasoning, limit arguments, and standard theorems of calculus.</p> <p>CO4: Construct recurrence relations, Taylor expansions, and mathematical models for real-life applications.</p>
9	<p style="text-align: center;">Syllabus</p> <p>UNIT I: Practical based on sequences and recurrence relations (30 Hours)</p> <ul style="list-style-type: none"> • Convergence of sequences of real numbers • Arithmetic of convergent sequences • Applications of sandwich theorem and divergent sequences • Homogeneous recurrence relations: degree one and two • Homogeneous recurrence relations: degree three and four • Non homogeneous recurrence relations • Examples of setting up recurrence relations <p>UNIT II: Practical based on Differentiability and Integrability of functions (30 Hours)</p> <ul style="list-style-type: none"> • Differentiability: examples from first principles • Arithmetic of differentiable functions • Derivate of the inverse of a function • Applications of Rolle's theorem and Mean value theorems • Taylor expansion of functions at a point • Applications of derivatives: maxima and minima • Applications of derivatives: further examples • Computations of simple integrals

Scheme of Examination and Assessment Pattern

Paper – 50 Marks

External Examination: Semester End External - 30 marks Time: 2:00 hour

Format of Question Paper

Question	Based On	Marks
Q1	Five out of Eight multiple choice questions (four from Unit 1 and four from Unit 2) (LO1 to LO3)	(3 × 5 = 15 Marks)
Q2	Attempt any Two out of Four (Two from Unit 1 and two From Unit 2). (LO 3 and LO 4)	(5 × 2 = 10 Marks)
Q3	Journal	5 (2.5 marks for each Unit 1 & Unit 2)
Total:		30

Note:

- **Certified Journal** is **compulsory** for appearing at the time of Practical Exam, failing which they will not be allowed to appear for the examination.
- Students are required to perform 75% of the Practicals for the journal to be duly certified. The journal serves as a record of their practical work and is essential component of the evaluation process.

Internal Examination: Continuous Evaluation - 20 marks

Continuous Evaluation through: Quizzes, Class Tests, presentations, projects, role play, creative writing, assignments etc. (at least 3)

	Assessment / Evaluation	Marks
1.	Objective question test	10
2.	Overall performance	5
3.	Viva	5

Total: 20

Paper pattern of the Test (Offline Mode):

Q1: (Attempt any 5 from 8) Multiple choice questions. (10 marks: 5×2)

Duration: 1Hrs

While setting question paper four MCQ on module 1 and four MCQ on module 2 both.

11

REFERENCES:

1. Bartle R. G. and Sherbert D. R., Introduction to Real Analysis, John Wiley and Sons.
2. Richard R. Goldberg, Methods of Real Analysis, John Wiley and Sons.
3. Thomas and Finney, Calculus and Analytical Geometry, Pearson
4. Ajit Kumar and S. Kumaresan, A basic course in real analysis, Chapman and Hall
5. B. V. Limaye and Sudhir Ghorpade, A course in calculus and real analysis, Springer Nature

**Smt. Chandibai Himathmal Mansukhani College
(Autonomous)**

**Third Year B. Sc.
(Computer Science)**

Semester- VI

Title: On Job Training (OJT)

**Vertical - 6
Continuing Education Program (OJT): 2 Credits**

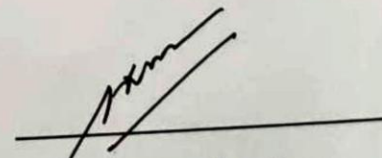
**with effect from
Academic Year 2026-2027**

Board of Studies in Computer Science:

Sr. No	Name of the Faculty	Designation and College
1.	Ms. Ritika Sachdev	Ad-hoc BoS Chairperson and Head of the Department, Smt. CHM College, Ulhasnagar.
2.	Mr. Amol Joglekar	Vice Chancellor Nominee
3.	Mr. Vipul Saluja	Assistant Professor, National College
4.	Mrs. Shalini Maheshgauri	Assistant Professor, KC College, HSNC University.
5.	Ms. Geeta Brijwani	Assistant Professor, KC College, HSNC University.
6.	Mr. Manoj Kevadia	Future Tech Ltd, Kalyan
7.	Ms. Neeta Rohra	Freelancer
8.	Ms. Lata Bhatia	Assistant Professor, Smt. CHM College, Ulhasnagar
9.	Ms. Madhvi Swami	Assistant Professor, Smt. CHM College, Ulhasnagar
10.	Ms. Gauri Hallale	Assistant Professor, Smt. CHM College, Ulhasnagar

Name & Signature of the Ad-hoc BoS
Chairperson:

Ms. Ritika Sachdev



Name & Signature of the Dean:

Mr. Agi Thomas

