Mahatma Education Society's

Pillai HOC College of Arts, Science & Commerce (Autonomous)

Rasayani

Affiliated to University of Mumbai
NAAC Accredited with "A+" Grade in cycle II
ISO 9001:2015 Certified

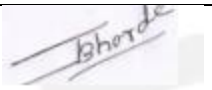
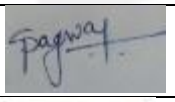
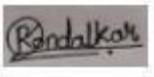
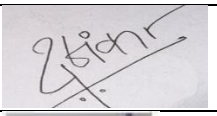
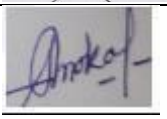
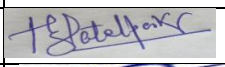
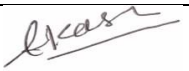
SYLLABUS

B.Sc. Computer Science
F. Y. B. Sc. Computer Science

As per National Education Policy 2020

Academic Year 2026-27

Mahatma Education Society's
College Code: 870
PILLAI HOC COLLEGE OF ARTS, SCIENCE & COMMERCE
Pillai HOCL Educational Campus, HOC Colony, Rasayani, Via. Panvel, Dist. Raigad. Pin 410207
Tel: 02192 - 669000 / 01 / 02 / 03 / 04 / 05 / 06 / 07 / 08 / 09 Website : www.phcasc.ac.in Email : phcasc@mes.ac.in
(NAAC Accredited 'A+' Grade , CGPA – 3.26 in Cycle 2 & ISO 9001:2015 Certified)
Affiliated to the University of Mumbai, Approved by Government of Maharashtra
(AUTONOMOUS COLLEGE)

Sr.No.	Name	Designation	Signature
1	Ms. Priyanka Sorte	Head of the Department of BSc in Computer Science	
2	Dr. Rinkoo Shantnu	Principal	
3	Mr. Binit Kumar	Vice -Principal	
4	Ms. Jyoti Borade	Member	
5	Ms. Sonali Dagwar	Member	
6	Ms. Rutuja Kondalkar	Member	
7	Mr. Yash Karkhanis	Member	
8	Ms. Aishwarya Mokal	Member	
9	Dr. Swapna Kadam	Vice Chancellor Nominee	
10	Dr. Annie Rajan	Subject Expert	
11	Dr. Homraj Patelpaik	Subject Expert	
12	Mr. Swapnil H. Patil	Industry Representative	
13	Mr. Akash Ghadge	Alumni Representative	

Introduction

A **B.Sc. in Computer Science** is a three-year undergraduate program designed to bridge the gap between theoretical knowledge and practical application, with a strong focus on developing a deep understanding of computer systems and networks. The comprehensive curriculum covers a wide range of software and hardware technologies and their real-world applications, enabling students to build strong problem-solving and program design skills while enhancing analytical thinking. Throughout the course, students gain technical proficiency by designing and developing computer programs in diverse and emerging areas such as web development, security, cloud computing, and data science, while applying core computer science principles and software engineering practices. The program also emphasizes innovation and professionalism by encouraging students to stay updated with industry trends, engage in lifelong learning, work effectively in teams, manage projects, and uphold ethical standards in the use of cyber systems. This interdisciplinary training prepares graduates for technical employment, higher education, entrepreneurship, and new career paths, supported by strong placement opportunities, incubation assistance, and exposure to modern advancements and emerging sub-fields in computer science.

Programme Outcomes (POs)

PO. No.	PO Title	POs in brief
PO1	Fundamental Knowledge Acquisition	Graduates will demonstrate a comprehensive and foundational knowledge of their chosen discipline along with an awareness of interdisciplinary connections.
PO2	Critical Thinking and Analytical Reasoning	Graduates will be able to analyse complex problems, synthesize data from multiple sources (qualitative and quantitative), and employ logical reasoning to formulate well-supported conclusions and arguments.
PO3	Effective Communication	Graduates will exhibit proficiency in both written and oral communication, articulating ideas clearly, persuasively, and ethically to diverse audiences
PO4	Problem Solving	Graduates will possess the ability to identify, formulate, and design solutions for real-world problems in their professional or social contexts, applying relevant theoretical knowledge and practical skills.
PO5	Information and Digital Literacy	Graduates will demonstrate the capability to locate, evaluate, and effectively use information from various sources, and utilize modern tools and Information and Communication Technology (ICT) for professional and academic tasks.
PO6	Research Skills and Scientific Temperament	Graduates will develop a sense of inquiry and research methodology, including the ability to design experiments (where applicable), collect and analyse data, and interpret results while maintaining scientific rigor and intellectual honesty.
PO7	Ethical Reasoning and Professional Integrity	Graduates will recognize ethical dilemmas, commit to professional and academic ethics, and demonstrate an understanding of moral and social responsibilities in their personal and professional conduct.
PO8	Employability and Professional Skills	Graduates will acquire the necessary job-ready skills, managerial competencies, and professional values to secure gainful employment or pursue advanced education in their respective fields.
PO9	Environmental and Sustainability Consciousness	Graduates will understand the importance of environmental conservation and sustainable development, displaying responsibility toward ecological challenges and advocating for healthy environmental practices.
PO10	Life-Long Learning	Graduates will develop the capacity for independent and self-directed learning to continuously upgrade their knowledge and skills, enabling them to adapt to rapid technological and societal changes.
PO11	Civic and Social Responsibility	Graduates will act as responsible citizens with an informed awareness of constitutional values, engaging proactively in community development and addressing social needs.
PO12	Empathy and Social Intelligence	Graduates will be able to cultivate and demonstrate affective, interpersonal, social and emotional intelligence.

Programme Specific Outcomes (PSOs)

PSOs. No.	PSO Title	PSOs in brief
PSO1	Technical Design & System Development	Graduates will be able to design, develop, and implement reliable computer applications and systems across diverse domains such as networking, web technologies, security, cloud computing, IoT, and data science.
PSO2	Application of Computing Principles & Engineering Practices	Graduates will apply core computer science theories, software engineering principles, tools, and technologies to analyze, model, and solve real-world computing problems efficiently.
PSO3	Innovation, Research Mindset & Professional Ethics	Graduates will demonstrate awareness of current technological trends, foster innovation in problem-solving, and uphold ethical standards in computing, Internet usage, and cyber practices.
PSO4	Teamwork, Project Management & Lifelong Career Growth	Graduates will work effectively as individuals and team members, manage software projects efficiently, and pursue continuous learning to excel in higher studies and evolving IT career opportunities.

Evaluation Pattern

Marking Code	Marking Scheme
A	50 Marks Semester End Exam, 50 Marks Continuous Assessment (distributed within 15 Marks Class Test, 15 Marks Presentation & Assignment, 10 Marks Online Quiz, 10 Marks Attendance & Class Participation)
B	50 Marks Semester End Exam
C	100 marks Continuous Assessment (distributed within 30 Marks Class Test, 30 Marks Presentation & Assignment, 30 Marks Online Quiz, 10 Attendance & Class Participation)
D	50 Marks of Continuous Assessment (distributed within 15 Marks Class Test, 15 Marks Presentation & Assignment, 10 Marks Online Quiz, 10 Marks Attendance & Class Participation)
E	50 Marks Practical Examination (distributed within 30 Marks Practical Module 1 & 2, 10 Marks Journal, 10 Marks Viva)

Course Structure

Semester I							
Course Code	**Course Type**	**Course Title**	**Theory/ Practical**	**Marks**	**Credits**	**Lectures/ Week**	**Evaluation Pattern**
HUSCS101	Major	Digital System and Architecture	Theory	100	2	2	A
HUSCS101P	Major - Practical	Practical (HUSCS101)	Practical	50	1	2	E
HUSCS102	Major	Database Management System	Theory	100	2	2	A
HUSCS102P	Major - Practical	Practical (HUSCS102)	Practical	50	1	2	E
HUSCS103	Minor	Descriptive Statistics	Theory	100	2	2	A
HUSCS103P	Minor - Practical	Practical (HUSCS103)	Practical	50	1	2	E
HUSCS104	SEC	Object Oriented Programming with C++	Theory	100	2	2	A
HUSCS104P	SEC - Practical	Practical (HUSCS104)	Practical	50	1	2	E
HBSCS105	IKS	Indian Roots & Information System	Theory	50	2	2	D
HUOE104	Multidisciplinary/ OE	Basics of Marketing Mix	Theory	100	3	3	C
HUAEC101	AEC	Communication Skills in English	Theory	50	2	2	D
HUVEC101	VAC/VEC	Fundamentals of Social and Emotional Skills	Theory	100	3	3	C
Total				**900**	**22**		**

Abbreviations:

SEC: Skill Enhancement Course
AEC: Ability Enhancement Course
VAC: Value Added Course
VEC: Value Education Course
IKS: Indian Knowledge System
OE: Open Elective

SEMESTER I

BOS	**Mathematics, Statistics and Computer Application**				
Course	**Digital System & Architecture**				
Course Code	**HUSCS101**	**Level**	**4.5**		
			Theory	**Practical**	**Total**
Semester	**I**	**Credits**	**2**	**1**	**3**
Type	**Major**	**No of Teaching hours**	**30**	**30**	**60**
Evaluation/ Assessment	**Total Marks**	**Semester End**	**Continuous**	**Practical**	
	150	**50**	**50**	**50**	

Learning Objectives	
1	Understand logic gates, number systems, and flip-flops through theoretical and practical exercises.
2	Analyze the structure and operation of digital systems and computers.
3	Explore processor architecture, memory systems, and control units.
4	Implement basic 8085 microprocessor programs and simulate digital circuits using HDL tools.SS
5	Understand emerging topics in modern architectures such as RISC-V, pipelining, and embedded system integration.

Course Outcomes	
CO1	Students will be able to apply number systems and digital codes in system design.
CO2	Students will be able to design and analyze basic combinational and sequential circuits using simulation tools.
CO3	Students will be able to understand memory hierarchy and cache coherence strategies.
CO4	Students will be able to write and simulate 8085 assembly programs.
CO5	Students will be able to explain modern processor trends including RISC, CISC, and embedded system applications.

Modules At Glance

Module No.	**Content**	**No. of Lectures**	**Mapping with CO**
1	Digital Logic & Computer System Overview	15	CO1, CO2, CO3
2	Memory & Microprocessor Systems	15	CO3, CO4, CO5
		30	

Syllabus

Module No.	Content	No. of Lectures
I	**Digital Logic & Computer System Overview** a. Fundamentals of Digital Logic: Boolean algebra, Logic Gates, Simplification of Logic Circuits: Algebraic Simplification, Karnaugh Maps. b. Combinational Circuits: Adders, Subtractors, Multiplexer, Demultiplexer. c. Sequential Circuits: Flip- Flops (SR, JK & D), Counters: synchronous and asynchronous Counter. Basic HDL Simulation (e.g., Logisim or VHDL introduction) d. 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. Simple real-world application case study (IoT/Embedded Systems)	15
II	**Memory & Microprocessor Systems** a. 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 b. Instructions: Instruction Formats, Instruction Sets, Instruction Cycle, Pipelining Addressing Modes, Addressing Modes Examples with Assembly Language c. Processor Organization: Structure and Function. Register Organization Basic Microprocessor operations: Data Transfer Operations, Arithmetic & Logical Operations,Introduction to RISC and CISC Architecture, RISC-V basics & Embedded Processor Applications, ARM vs. x86 comparison – industry relevance d. Microcontroller: Brain of Smart Device, Difference between Microprocessor & Microcontroller, Applications of Microcontroller	15
Case Study Scenario		
1	A hospital wants to design a real-time patient monitoring system that collects data (heart rate, temperature, oxygen level) and sends alerts when readings cross safe limits. The system uses an 8085 microprocessor for basic processing and cache memory for faster access**.**	
2	A smartphone company is deciding whether to use ARM Cortex series or Intel x86 processors. They aim for high battery life, compact design,and strong performance.	

References Books

- M. Mano – Computer System Architecture, 3rd Ed., Pearson
- Carl Hamacher – Computer Organization and Embedded Systems, 6th Ed., McGraw-Hill
- R.P. Jain – Modern Digital Electronics, 4th Ed., Tata McGraw Hill
- Frank Vahid – Digital Design with RTL, VHDL, and Verilog
- John Paul Shen – Modern Processor Design

Semester End Evaluation (50 Marks) ***Time: 2 Hr***

Paper Pattern

Question No	**Questions**	**Total Marks: 50**
Q1	Attempt any 3 out of 5	15
Q2	Attempt any 3 out of 5	15
Q3	Attempt any 3 out of 5	15
Q4	Case Study	05

Practical Syllabus

Sr. No.	List of Practical	Mapped to CO	No of Lectures
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.	CO1	3
2	**Boolean Expression Simplification:** Simplify given Boolean expressions and realize them using Logisim.	CO2	3
3	**Half/Full Adder Design:** Design and verify the operation of a half/full adder using Logisim.	CO3	3
4	**Half/Full Subtractor Design:** Design and verify the operation of a half/full subtractor using Logisim.	CO3	3
5	**4-Bit Magnitude Comparator:** Design a 4-bit magnitude comparator using combinational circuits in Logisim**.**	CO3	3
6	**Flip-Flop Implementation:** Verify the operation of flip-flops (e.g., D, JK) using logic gates in Logisim.	CO4	3
7	**Counter Operation Verification:** Verify the operation of a counter using Logisim.	CO4	3
8	**4-Bit Shift Register Operation:** Verify the operation of a 4-bit shift register using Logisim.	CO5	3
9	**Multiplexer/Demultiplexer Design:** Design and implement expressions using multiplexers/demultiplexers in Logisim**.**	CO5	3
10	**I3-Bit Binary Ripple Counter:** Design and implement a 3-bit binary ripple counter using JK flip-flops in Logisim.	CO5	3

Practical Evaluation (50 Marks)

Question No	Assessment/ Evaluation	Marks
Q1	Practical (Module 1 & Module 2)	30
Q2	Journal	10
Q3	Viva and Attendance	10

BOS	**Mathematics, Statistics and Computer Application**				
Course	**Database Management System**				
Course Code	**HUSCS102**	**Level**	**4.5**		
			Theory	**Practical**	**Total**
Semester	**I**	**Credits**	2	1	3
Type	**Major: Mandatory**	**No of Teaching hours**	30	30	60

Evaluation/ Assessment	**Total Marks**	**Semester End**	**Continuous**	**Practical**
	150	**50**	**50**	**50**

Learning Objectives	
1	To make students aware of the fundamentals of the database system.
2	To give an idea how ERD components are helpful in database design and implementation.
3	To experience the students working with databases using MySQL.
4	To familiarize the student with normalization, database protection and different DCL Statements.
5	To make students aware about the importance of protecting data from unauthorized users.
6	To introduce students to the concept and purpose of indexing in database systems.
7	To make students aware of granting and revoking rights of data manipulation.

Course Outcomes	
	After successful completion of this course, students would be able to: -
CO1	To appreciate the importance of database design.
CO2	Analyze database requirements and determine the entities involved in the system and them relationship to one another.
CO3	Write simple queries to MySQL related to String, Math's and Date Functions.
CO4	Create tables and insert/update/delete data, and query data in a relational using MySQL commands.
CO5	Understand the normalization and its role in the database design process. Handle data permissions.
CO6	Create indexes and understand the role of Indexes in optimization search.

Modules At Glance

Module No.	**Content**	**No. of Lectures**	**Mapping with CO**
1	Fundamentals of Database Concepts and Data Modeling.	15	CO1, CO2, CO5
2	Advanced Concepts in Relational Databases and Database Management.	15	CO3, CO4, CO5, CO6
	Total	30	

Syllabus

Module No.	Content	No. of Lectures
I	**Fundamentals of Database Concepts and Data Modeling.** 1. **Introduction to DBMS:** Database, DBMS – Definition, Overview of DBMS,Advantages of DBMS, Levels of abstraction, Data independence, DBMSArchitecture. 2. **Data models:** Client/Server Architecture, Object Based Logical Model, RecordBased Logical Model (relational, hierarchical, network). Relational data model**:** Domains, attributes, Tuples and Relations 3. **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 beyondER) Entity to Table, Relationship to tables with and without key constraints. 4. **DDL Statements:** Creating Databases, Using Databases, data types, Creating Tables(with integrity constraints – primary key, default, check, not null), Altering Tables,Renaming Tables, Dropping Tables, Truncating Tables. 5. **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	15
II	**Advanced Concepts in Relational Databases and Database Management.** 1. **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) 2. **Joining Queries:** inner join, outer join (left outer, right outer, full outer). Views**:** Creating, altering, dropping, renaming and manipulating views. 3. **Normal forms:** Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition. 4. **DCL Statements:** Creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges), Transaction control commands –Commit, Rollback. 5. **Database Protection:** Security Issues, Security Mechanisms Discretionary Access Control, Backing Up and Restoring databases, Threats to Database.**Database Administration and Security:** Data as a Corporate Asset, The Need for a Database and	15

	Its Role in an Organization, The DBA's Role.	
Case Study Scenario		
I	University Course Registration System:A large university wants to design a database for its Course Registration System. The university needs to track students, instructors, courses, departments, classrooms, and student registrations. The system must also support rules such as prerequisite constraints, teaching assignments, and scheduling.	
II	Online Bookstore Management System:An online bookstore, BookVerse, wants to analyze sales, customers, authors, and books. The database stores information about books, authors, customers, and orders. Managers frequently request reports, so the business wants to use joins and views to simplify querying.	

References Books

- Database System Concepts, Abraham Silberschatz, HenryF.Korth, S.Sudarshan, McGraw Hill,2017
- MySQL: The Complete Reference, VikramVaswani , McGraw Hill, 2017
- Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease, Ashwin Pajankar, BPB Publications, 2020
- Fundamentals of Database Systems, Elmasri Ramez and Navathe Shamkant B, Pearson Education 6th Edition, 2010.
- Database System Concepts Silberschatz, Korth, Sudarshan, McGraw Hill, 5 Edition, 2006.

Semester End Evaluation (50 Marks) ***Time: 2 Hr***

Paper Pattern

Question No	**Questions**	**Total Marks: 50**
Q1	Attempt 3 out of 5	**15**
Q2	Attempt 3 out of 5	**15**
Q3	Attempt 3 out of 5	**15**
Q4	Case Study	**05**

Practical Syllabus

Sr. No	List of Practical	CO Mapping	No. of Lectures
1	**To design an ER model for an Employee Payroll System to represent employee details, payroll processing, and related relationships.** a. Identifying and listing all required entities b. Defining attributes for each entity c. Establishing relationships among entities using ER notations d. Drawing the ER diagram to represent the system structure visually e. Verifying the model for normalization and data consistency	CO1, CO2	3
2	**Write SQL query for given problem statement.** a. Viewing all databases b. Creating a Database c. Viewing all Tables in a Database	CO3, CO4	3
3	**Perform the following Operations.** a. Creating Tables (With and Without Constraints) b. Inserting/Updating/Deleting Records in a Table c. Saving (Commit) and Undoing (rollback)	CO3, CO4	3
4	**Perform the following Operations.** a. Altering a Table b. Dropping/Truncating/Renaming Tables c. Backing up / Restoring a Database.	CO4	3
5	**Perform following:** a. Simple Queries with Where Operators b. Where with Keywords and Logical Operators c. Simple Queries with Aggregate functions d. Queries with Aggregate functions (group by and having clause)	CO3	3
6	**Perform Queries involving:** a. Date Functions b. String Functions c. Math Functions	CO3	3
7	**Retrieving Data from Multiple Table:** a. Joining Tables (InnerJoins, Outer-Joins) b. Aliases for Table Names	CO3, CO4	3
8	**Perform Views commands:** a. Creating Views b. Dropping Views c.Selecting from view	CO4	3

9	**Perform DCL statements:** a. Granting permissions b. Revoking permissions	CO5	3
10	**Use of DDL DML and DCL statement for employee payroll system (Or any other system recommended by teacher)** a. Identify employee payroll entities b. Define and Create Database, tables with constraint c. Insert 10 to 20 relevant records to support further queries d. Design simple queries using operators and functions e. Create Views and perform drop and select view command Perform grant and revoke commands.	CO1, CO2,CO 3, CO4, CO5, CO6	3

Notes: The Practicals will be performed in PostgreSQL/Oracle

Practical Evaluation (50 Marks)

Question No	Assessment/ Evaluation	Marks
Q1	Practical (Module 1 & Module 2)	30
Q2	Journal	10
Q3	Viva and Attendance	10

BOS	**Mathematics, Statistics and Computer Application**				
Course	**Descriptive Statistics**				
Course Code	**HUSCS103**	**Level**	4.5		
			Theory	**Practical**	**Total**
Semester	**I**	**Credits**	2	1	3
Type	**Minor**	**No of Teaching hours**	30	30	60

Evaluation/ Assessment	**Total Marks**	**Semester End**	**Continuous**	**Practical**
	150	50	50	50

Learning Objectives	
1	To understand different types of Data, and to analyze and present the data
2	To compute various Measures of Central Tendencies.
3	To compute various Measures of Dispersion.
4	To understand the concept of Skewness and Kurtosis.
5	To compute Correlation Coefficient for bivariate data and further apply the regression analysis.

Course Outcomes	
CO1	Able to organize, manage and present the data.
CO2	To understand the use Measures of Central Tendencies and Dispersion.
CO3	Able to understand and compute the consistent and inconsistent data
CO4	Able to identify the association between variables
CO5	Able to understand forecasting techniques and to find cause and effect relationship between variable through regression analysis.

Modules At Glance

Module No.	Content	No. of Lectures	Mapping with CO
I	1.1 Introduction of Statistics: 1.2. Measures of Central Tendencies: 1.3. Measures of Dispersion: 1.4. Raw and Central Moments, relation between Raw and Central moments, concept of Skewness and Kurtosis.	15	CO1, CO2, CO3
II	2.1. Correlation 2.2. Rank Correlation 2.3. Regression, 2.4 Relation between Correlation and Regression 2.5. Concept of multiple correlation 2.6. Concept of multiple regression and logistics regression	15	CO4, CO5
		30	

Syllabus

Module No.	Content	No. of Lectures
I	1. Introduction of Statistics: Meaning of Statistics, Importance of Statistics, Types of Characteristics, Different types of Scales: Nominal, Ordinal, Interval and ratio. Univariate frequency distribution of discrete and continuous variables and Cumulative frequency distribution. Data Presentation: Frequency Distribution, Histogram and Ogives Curves. 2. Measures of Central Tendencies: Concept of Central Tendency, characteristics of good measures of Central Tendency, Positional Averages: Median, Mode , Partition values: Quartiles, Deciles and Percentiles -examples of ungrouped and grouped data 3. Measures of Dispersion: Concept of Dispersion, Requirements of good measures of Dispersion, Absolute and Relative measures of Dispersion: Range, Quartile Deviation, Mean Absolute Deviation, Standard Deviation, Combined Standard Deviation-examples of ungrouped and grouped data 4. Raw and Central Moments, relation between Raw and Central moments, concept of Skewness and Kurtosis.	15
II	1. Concept of Correlation, types and interpretation, Scatter Diagram, Product Moment Correlation Coefficient, and its properties 2. Spearman' s Rank Correlation (with and without ties) 3. Concept of Linear Regression, Principle of Least Square, Fitting a straight line by method of least square. 4. Difference between Correlation and Regression, relation between Correlation and Regression 5. Concept of multiple correlation 6. Concept of multiple regression and logistics regression	

Case Study Scenario	
1	A retail company collects data on customer age, product type and monthly spending to understand buying behavior. The data is classified using appropriate scales and organized into frequency and cumulative frequency distributions. Graphs such as histograms and ogives are used for presentation, and measures of central tendency like mean, median, mode, quartiles and percentiles are applied to identify average spending patterns and compare different customer groups.
2	A college analyzes the performance of students by studying the relationship between study hours and marks. Measures of dispersion and moments are used to understand variability and distribution shape. Correlation and regression techniques are applied to determine and predict the relationship between variables, including the effect of multiple factors such as attendance and internal marks on overall academic performance.

References Books:

- Agarwal, B. L. (2003). Programmed statistics (2nd ed.). New Age International Publishers.
- Purohit, S. G., Gore, S. D., & Deshmukh, S. R. (2008). Statistics using R. Narosa Publishing House.
- Sarma, K. V. S. (2001). Statistics made it simple: Do it yourself on PC. Prentice Hall of India.
- Stephens, L. J. (2000). Schaum's outline of theory and problems of beginning statistics. McGraw-Hill.
- Montgomery, D. C., & Runger, G. C. (2014). Applied statistics and probability for engineers (6th ed.). John Wiley & Sons.

Semester End Evaluation (50 Marks) ***Time: 2 Hr***

Paper Pattern

Question No	Questions	Total Marks: 50
Q1	Attempt 3 out of 5	**15**
Q2	Attempt 3 out of 5	**15**
Q3	Attempt 3 out of 5	**15**
Q4	Case Study	**05**

Practical Syllabus

Sr. No	List of Practical	Mapping with CO	No. of Lectures
1.	**Introduction to Excel** a. Understanding Data Tools. b. Understanding Formula Tools, insert functional library using insert function. c. Add-Ins Analysis tool packs d. Formula writing, Functions, using Cell reference, Sort, Filter and Advance Filter	CO1	3
2.	**Data Entry and Manipulation** a. Tools for data entry and accuracy: Quick Access toolbar customization, Form tool b. Data Transposition to fit Excel (as an Array) c. Data Conversion with the Logical If, VLOOKUP, Pivot table, Pivot chart. d. Data conversion to Text from Non-Excel Sources. Using text to Column	CO1	3
3.	**Data Validation** a. Specifying a valid range of values for a cell. b. Specifying a list of valid values for a cell c. Specifying custom validation based on formula for a cell.	CO1	3
4.	**Diagrams and Graphs** a. Excel program to plot Histogram. b. Excel program to plot Bar Graphs. c. Excel program to plot Pie chart. d. Boxplot and multiple Boxplots	CO1	3
5.	**Measures of Central Tendency** a. Excel program to find Mean. b. Excel program to find Median. c. Excel program to find Mode	CO2	3
6.	**Measures of Dispersion** a. Excel program to find Range, Interquartile Range b. Excel program to find Variance. c. Excel program to find Standard Deviation. d. Excel program to find Skewness and Kurtosis..	CO3	3
7.	**Correlation** a. Excel program to find Positive Correlation. b. Excel program to find Negative Correlation. c. Excel program to find Zero Correlation	CO4	3
8.	**Regression-1** a. Excel program to perform linear regression for prediction. b. Excel program to perform polynomial regression for prediction	CO5	3
9.	**Regression – 2** a. Excel program to perform multiple linear regression for prediction. b. Excel program to perform logistic regression for prediction	CO5	3

10.	Design a survey form, get primary data and analyse it. (given case study)	CO1,CO2,CO3,CO14,CO5	3

Practical Evaluation (50 Marks)

	Assessment/ Evaluation	Marks
Q1	Practical (Module 1 & Module 2)	30
Q2	Journal	10
Q3	Viva and Attendance	10

BOS	**Mathematics, Statistics and Computer Application**				
Course	**Object Oriented Programming with C++**				
Course Code	**HUSCS104**	**Level**	**4.5**		
			Theory	**Practical**	**Total**
Semester	**I**	**Credits**	2	1	3
Type	**SEC**	**No of Teaching hours**	**30**	**30**	**60**

Evaluation/ Assessment	**Total Marks**	**Semester End**	**Continuous**	**Practical**
	150	**50**	**50**	**50**

Learning Objectives	
1	Understand and apply the basic concepts and features of object-oriented programming in C++, including classes, objects, constructors, and destructors.
2	Use decision-making structures, loops, arrays, and strings to control program flow and manage data efficiently.
3	Demonstrate the use of function, constructor, and operator overloading to implement compile-time polymorphism.
4	Work with pointers and inheritance to build modular and reusable code using various access specifiers.
5	Perform file operations using file handling techniques for both text and binary files.
6	Implement exception handling to create robust and error-tolerant C++ programs.

Course Outcomes	
CO1	Students are able to understand the fundamental principles of object-oriented programming and the structure of C++ programs.
CO2	Students are able to develop C++ programs using data types, operators, control structures, arrays, and strings.
CO3	Students are able to implement classes, objects, constructors, destructors, and apply appropriate access specifiers.
CO4	Students are able to apply advanced concepts such as inheritance, polymorphism, and pointers to build modular programs.
CO5	Learners will be able to demonstrate file handling and exception handling mechanisms to develop reliable and efficient applications.

Modules At Glance

Module No.	Content	No. of Lectures	Mapping with CO
1	Fundamentals of Programming	15	CO1, CO2, CO3
2	Advanced Programming Concepts	15	CO4, CO5
		30	

Syllabus

Module No.	Content	No. of Lectures
I	**Fundamentals of Programming** 1. **Introduction to Programming Concepts:** Object oriented programming paradigm, basic concepts of object-oriented programming, benefits of object-oriented programming, applications of object-oriented programming. Tokens, features of C++ and its basic structure, 2. **Data Types, Data Input Output and Operators:** Basic data types, variables, programming constants, cout and cin statements, operators 3. **Decision Making, Loops, Arrays and Strings:** Conditional statements-if, if...else, switch loops- while, do...while, for, types of arrays and string. 4. **Classes & Object:** Classes and objects, Dot Operator, data members, member functions, passing data to functions, scope and visibility of variables in function, Access Specifier. 5. **Constructors and Destructors:** Default constructor, parameterized constructor, copy constructor, private constructor, destructors.	15
II	**Advanced Programming Concepts** 6. **Polymorphism**: Binding-static binding & overloading, constructor overloading function overloading, operator overloading, Runtime Polymorphism. 7. **Pointers:** Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes 8. **Inheritance:** Defining base class and its derived class, access specifiers, types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance, friend function and friend class, constructors in derived classes. 9. **File Handling:** File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling. 10. **Exception Handling:** Exception Handling: Introduction, Advantages of Exception Handling, Exception Handling Mechanism, Concept of throw & catch with example.	15

Case Study Scenario	
I	Smart Fitness Tracker Application A tech startup is building a Smart Fitness Tracker App called FitLife, which monitors user activities, calculates workout statistics, stores data in arrays, and processes user input strings. The development team must implement several core features using decision-making statements, loops, arrays, and string processing.
II	Online Payment Processing System A company is developing an online payment gateway called **PayMate**. The system allows customers to: Enter payment details Validate card number Process transactions Handle network or server failures Confirm payment success Because payment systems must be highly reliable, the application needs

References Books

- **Mastering C++**, K. Venugopal, Rajkumar Buyya, T. Ravishankar, Publisher: McGraw-Hill Education, 2013.
- **Object Oriented Programming with C++**, E. Balagurusamy, Publisher: McGraw-Hill Education, 2017.
- **The C++ Programming Language**, Bjarne Stroustrup, Publisher: Addison-Wesley, 2013.
- **C++: The Complete Reference**, Herbert Schildt, Publisher: Tata McGraw-Hill, 2003.
- **Object-Oriented Programming in C++**, Robert Lafore, Publisher: Pearson Education, 2008.

Semester End Evaluation (50 Marks) ***Time: 2 Hr***

Paper Pattern

Question No	Questions	Total Marks: 50
Q1	Attempt 3 out of 5	15
Q2	Attempt 3 out of 5	15
Q3	Attempt 3 out of 5	15
Q4	Case Study	05

List of Practical

Sr. No.	List of Practical	CO Mapping	No. of Lectures
1	**Practical 1: Basic C++ Program Structure** a. Write a program to display "Hello, World!". b. Use #include <iostream>, main () function, and return 0; c. Compile and run using command line (g++) or IDE.	CO1	3
2	**Practical 2: Using Data Types and Operators** a. Declare and initialize int, float, char, bool. b. Write a calculator for +, -, *, /, %. c. Use logical operators (&&, ||, !) in conditional statements.	CO2	3
3	**Practical 3: Conditional Statements and Loops** a. Check whether a number is even or odd using if...else. b. Menu-driven program using switch c. Use for, while, and do...while to: i. Print a triangle pattern ii. Find the sum of first N numbers	CO2	3
4	**Practical 4: Arrays and String Manipulation** a. Declare and display elements of a 1D array b. Sort and search elements in an array c. Use character arrays to reverse a string	CO2	3
5	**Practical 5: Constructors and Destructors** a. Create a class with a default constructor to initialize values. b. Use parameterized constructor to accept data. c. Implement a copy constructor. d. Add a destructor to show object cleanup.	CO3	3
6	**Practical 6: Function Overloading, Operator Overloading, and Overriding** a. Create overloaded functions (area() for circle, rectangle). b. Overload + operator to add two complex numbers. c. Overload == to compare two student objects. d. Implement base class with virtual function and override it in derived class.	CO3	3
7	**Practical 7: Using Pointers in C++** a. Declare an integer pointer and modify value via pointer. b. Use pointer to access an array. c. Dynamically allocate memory using new. d. Deallocate memory using delete.	CO4	3
8	**Practical 8: Implementing Inheritance** a. Write a C++ Program that illustrate single inheritance. b. Write a C++ Program that illustrate multiple inheritance. c. Write a C++ Program that illustrate multi-level	CO4	3

	inheritance. d. Write a C++ Program that illustrate Hierarchical inheritance.		
9	**Practical 9: File Handling Operations** a. Write to a file using ofstream. b. Read from a file using ifstream. c. Create a binary file for student records (roll, name, marks). d. Display contents from the binary file.	CO5	3
10	**Practical 10: Exception Handling** a. Demonstrate try, throw, and catch with divide-by-zero. b. Use multiple catch blocks (int, char, string). c. Validate input (e.g., negative age) and throw custom exceptions. d. Catch all exceptions using catch(...).	CO5	3

Practical Evaluation (50 Marks)

	Assessment/ Evaluation	Marks
Q1	Practical (Module 1 & Module 2)	30
Q2	Journal	10
Q3	Viva and Attendance	10

BOS	**Mathematics, Statistics and Computer Application**				
Course	**Indian Roots of Information Science**				
Course Code	**HBSCS105**	**Level**	**4.5**		
			Theory	**Total**	
Semester	**I**	**Credits**	02	02	
Type	**IKS**	**No of Teaching hours**	**30**	**30**	
Evaluation/ Assessment	**Total Marks**	**Semester End**	**Continuous**	**Practical**	
	50	**NA**	**50**	**NA**	

Learning Objectives	
1	To sensitize the students about context in which they are embedded i.e. Indian culture and civilisation including its Knowledge System and Tradition.
2	To help students to understand the knowledge, art and creative practices, skills and values in the ancient Indian system.
3	To introduce the contribution from Ancient Indian system & tradition to modern science & Technology.

Course Outcomes	
CO1	Learner will understand and appreciate the rich Indian Knowledge Tradition.
CO2	Learner will understand the contribution of Indians in various fields
CO3	Learner will experience increase subject-awareness and self-esteem

Modules At Glance

Module No.	**Content**	**No. of Hours**	**Mapping with CO**
1	Introduction to Indian Knowledge System	15	CO1,CO2
2	Basics Applications inspired by IKS	15	CO2,CO3
		30	

Syllabus

Module No	Content	No. of Hours
I	**Introduction to Indian Knowledge System:** • Introduction to IKS: What is knowledge System, Characteristic & Features of Indian Knowledge System • Why IKS?: Cultural Awareness and Pride,Holistic Learning,Preservation and Innovation • Tradition & Scope of IKS : Vedas and Upanishads,Mathematics in Ancient India, Architecture & Vāstu, Scope: Natural Language Processing (NLP),Artificial Intelligence & Machine Learning,Algorithm Optimization Inspired by Vedic Math • History of Computers: Generations of Computers & Types of Computers	15
II	**Basic Applications inspired by IKS** • Mathematics & Computation : Logic & Reasoning (Nyāya and Anumāna) Binary Concepts using Vedic Mathematics • Patterns & Algorithms (Chandaḥśāstra – Prosody) **&** Language Structure (Paninian Grammar):Patterns in poetic meters → similar to binary sequences, Application in text analysis and compression algorithms, Concepts of meta-rules → foundations of programming languages , Influence on Natural Language Processing (NLP) • Time Representation & Calendars (Panchanga): Ancient Indian calendars use complex astronomical data, Concepts of time, cycles → used in simulations, data modeling • Classification and Categories (Ayurvedic Ontologies): Tridosa and Disease Classification,Knowledge Management,Data Analysis,Tools and Technologies	15

References Books

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 Banrasidas, 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