MCA SCHEME & SYLLABUS

(2025-2026)

B.V.V.S

BVVS

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102

Department of Computer Applications (MCA)

For the Admitted Batch 2025-26

MCA–I Semester Scheme of teaching and examinations for 2025-2026

Sl. No.	Category	Course Code	Course Name	Credits	Hours/Week			Examination Marks		
					Lecture	Tutorial /SDA	Practical/ Seminar	CIE	SEE	Total
1	IPCC	1PCAA101C	Programming and Problem Solving in C	4	3	-	2	50	50	100
2	BSC	1PCAA102C	Mathematical Foundation for Computer Applications	3	3	-	-	50	50	100
3	PCC	1PCAA103C	Database Management Systems	3	3	-	-	50	50	100
4	PCC	1PCAA104C	Operating Systems	3	3	-	-	50	50	100
5	PCC	1PCAA105C	Computer Networks	3	3	-	-	50	50	100
6	PCCL	1PCAA106L	DBMS Laboratory	2	-	-	4	50	50	100
7	PCCL	1 PCAA107L	Web Technologies Laboratory	2	-	-	4	50	50	100
8	**NCMC**	**1PCAA001M**	**Bridge Course Mathematics for MCA Program**	PP	3	-	-	100	-	100
			Total	**20**	**18**	**00**	**10**	**450**	**350**	**800**

Note:**BSC**-BasicScienceCourses,**PCC**:Professionalcore.**IPCC**-IntegratedProfessionalCoreCourses,**PCC(PB):**ProfessionalCoreCourses(ProjectBased),**PCCL**-ProfessionalCoreCourselab**,NCMC**-NoneCreditMandatoryCourse,**PP-**Pass/Pass, **NP-** Not Pass ,**L**-Lecture**,P**-Practical**,T/SDA-**Tutorial**/**SkillDevelopmentActivities(Hours are for Interaction between faculty and students NCMC-Research Methodology (**Online**) course is not counted for vertical progression; Students have to qualify for the award of the master's degree.

NCMC- Non Credit Mandatory Course (NCMC),Research Methodology, which students have to register and pass the course mandatorily through NPTEL/VTU Online Courses between 1st to 3rd semester for the award of the degree. Passing the course will be as per NPTEL/VTU Online Course Criteria. The result of the same will be reflected in the 4th semester.

MOOCS (3 credits / 12 weeks- PEC/MDC): This is mandatory course of 3 credits (12 weeks duration), which student have to register between 1st to 3rd semester or 4th semester from the BOS approved list of Swayam/ NPTEL /VTU Online Courses and passed with minimum of **50%** score for the award of the degree. The result will be reflected in the 4th semester.

Bridge Course: Non-Credit Mandatory Course - Bridge Course Mathematics for MCA Program Students who have not taken Mathematics at the 10+2 or degree level are required to study and pass this course in the 1st semester. However, this course/subject will not be considered for vertical progression.

BSC: Basic Science Courses: Courses like Mathematics/ Science are the prerequisite courses that the concerned board of Studies will decide.

PCC: Professional Core Course: Courses related to the stream of engineering, which will have both CIE and SEE components, students have to qualify in the course for the award of the degree. **Integrated Professional Core Course (IPCC)**: Refers to a Professional Theory Core Course Integrated with practical of the same course. The IPCC's theory part shall be evaluated by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE).However, questions from the practical part of IPCC shall be included in the SEE question paper. **PCCL: Professional Core Course Laboratory:** Practical courses whose CIE will be evaluated by the class teacher and SEE will be evaluated by the two examiners.

B.V.V.S

BVVS

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102

Department of Computer Applications (MCA)

For the Admitted Batch 2025-26

MCA-II Semester Scheme of teaching and examinations for 2025-2026

Sl. No.	Category	Course Code	Course Name	Credits	Hours/Week			Examination Marks		
					Lecture	Tutorial / SDA	Practical	CIE	SEE	Total
1.	IPCC	1PCAA201C	Introduction to Python	4	3	-	2	50	50	100
2.	PCC	1PCAA202C	Object Oriented Programming using JAVA	3	3	-	-	50	50	100
3.	PCC	1PCAA203C	Data Structures and Algorithms	4	4	-	-	50	50	100
4.	PCC	1PCAA204C	Software Engineering	3	3	-	-	50	50	100
5.	PCC	1PCAA205C	Introduction to AI	3	3	-	-	50	50	100
6	PCCL	1PCAA206L	Object Oriented Programming using JAVA Laboratory	1.5	-	-	3	50	50	100
7.	PCCL	1PCAA207L	Data Structures and Algorithms Laboratory	1.5	-	-	3	50	50	100
8.	**NCMC**	**1PCA A002M**	Seminar	PP	-	-	-	100	-	100
			Total	**20**	**16**	**00**	**08**	450	350	800

Note: **BSC**-Basic Science Courses, **PCC:** Professional Core Course**. IPCC**-Integrated Professional Core Courses, **PCC(PB):** Professional Core Courses (Project Based), **PCCL**-Professional Core Course lab ,**NCMC**- Non Credit Mandatory Course, ,L-Lecture, P-Practical, T/SDA-Tutorial / Skill Development Activities(Hours are for Interaction between faculty and students).

BSC: Basic Science Courses: Courses like Mathematics/ Science are the prerequisite courses that the concerned board of Studies will decide. **PCC:** Professional Core Course: Courses related to the stream of engineering, which will have both CIE and SEE components, students have to qualify in the course for the award of the degree**. Integrated Professional Core Course (IPCC):** Refers to a Professional Theory Core Course Integrated with practical of the same course. The IPCC's theory part shall be evaluated by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from the practical part of IPCC shall be included in the SEE question paper. **PCCL:** Professional Core Course Laboratory: Practical courses whose CIE will be evaluated by the class teacher and SEE will be evaluated by the two examiners. **Seminar -** Non Credit Mandatory Course (NCMC), Students have to select the relevant topic and prepare on the same and present the content in the class. Student has to submit seminar report to the department. The Head of the Department shall make an arrangement for conducting seminars through concerned faculty members of the Department. The committee, constituted for the purpose by the Head of the Department, shall award only the CIE marks for the seminar. The committee comprising of Guide/Co-Guide, Senior faculty of the department and HoD/HoD nominee.

B.V.V.S

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102

Department of Computer Applications (MCA)

For the Admitted Batch 2025-26

BVVS

MCA-III Semester Scheme of teaching and examinations for 2026-2027

Sl. No.	Course	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
					Lecture	Tutorial / SDA	Practical	CIE	SEE	Total
1.	PEC	1PCAA3XXE	Specializations	3	3	-	-	50	50	100
2.	PEC	1PCAA3XXE	Specializations	3	3	-	-	50	50	100
3.	PEC	1PCAA3XXE	Specializations	3	3	-	-	50	50	100
4.	PROJ	1PCAA3XXP	Project Work	15	25-30 hours per week			50	50	100
			Total	**24**	**09**	**00**	**00**	200	200	400

Students can opt any three courses from any one specialization listed below:

Specializations

Specialization	**AI and Data Science**	**Specialization**	**Network and System Administration**	**Specialization**	**S/W Systems and Web Application Development**
Course Code	**Course Title**	**Course Code**	**Course Title**	**Course Code**	**Course Title**
1PCAB301E	Data Mining and Data Warehousing	1PCAC301E	Block Chain Technology	1PCAD301E	Full Stack Development using Java
1PCAB302E	Big data Analytics	1PCAC302E	Network and Linux Administration	1PCAD302E	Rich Internet Application Development
1PCAB303E	Introduction to Generative AI	1PCAC303E	Unix Shell Programming	1PCAD303E	Object-oriented Analysis & Design
1PCAB304E	Artificial Neural Networks	1PCAC304E	Cloud Essentials	1PCAD304E	Software Architectures
1PCAB305E	Natural Language Processing	1PCAC305E	Cyber Security	1PCAD305E	Computer Organization and Software Systems
1PCAB306E	Deep Learning Fundamentals	1PCAC306E	Cryptography and Network Security	1PCAD306E	Management Information Systems

BVVS

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102
Department of Computer Applications (MCA)
For the Admitted Batch 2025-26

Specialization	**Cloud Technology**	**Specialization**	**IoT**
Course Code	**Course Title**	**Course Code**	**Course Title**
1PCAE301E	Cloud Computing	1PCAF301E	Data Management for IoT
1PCAE302E	Devops	1PCAF302E	Networked Embedded Applications
1PCAE303E	Data Storage Technologies and Networks	1PCAF303E	Cross Platform Application Development
1PCAE304E	Design and Operation of Data Centres	1PCAF304E	IoT Technology and Applications
1PCAE305E	Mobile Application Development	1PCAF305E	Communication and Networking Technologies in IoT
1PCAE306E	Wireless and Mobile Computing	1PCAF306E	Software and Programming in IoT

Project work is a significant component aimed at fostering research, practical application of knowledge, and innovation. The evaluation process generally follows these steps:

1. **Selection and Approval of Project Work:**
   - Topic Selection: Students propose project topics, often in consultation with their faculty advisor.
   - Approval Process: The proposed topic is submitted for approval by a project committee or department, ensuring alignment with academic standards and relevance.

2. **Project Execution:**
   - Research and Development: Students carry out research, experiments, or development work as per the project plan.
   - Periodic Reviews: Regular progress reviews are conducted by faculty to monitor the project's progress and provide feedback.
   - Documentation: Students maintain a detailed record of their methodology, data, results, and analysis. Finalized Scheme V4 8

3. **Submission of the Project Report:**
   - Format and Guidelines: The report must follow the prescribed format by the university or department.
   - Plagiarism Check: The report is often checked for plagiarism to ensure originality.

4. **Evaluation Process:**
   - Internal Evaluation: Faculty members from the department review the project report and presentation for content quality, innovation, and depth of research.
   - External Evaluation: An external examiner, often an industry expert or academician from another institution, reviews the project.
   - Viva Voce Examination: The student defends their project work before a panel comprising internal and external examiners. This assesses their understanding, analytical ability, and application of the project work.

BVVS

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102
Department of Computer Applications (MCA)
For the Admitted Batch 2025-26

5. Grading Criteria (Guidelines only)

- Report Quality: Depth of research, organization, and clarity of the document.
- Presentation Skills: Effectiveness in communicating key aspects of the project.
- Technical Merit: Innovation, accuracy, and the applicability of the research.
- Viva Performance: Understanding of the subject, responses to questions, and ability to discuss the work effectively.

6. Final Outcome:

- Marks Allocation: Typically, evaluation is a blend of internal (guided by the department) and external (examiner's input) assessments, distributed over the report, presentation, and viva.
- Pass Requirement: Students must meet a minimum threshold to pass, as per university policies.

This structured evaluation ensures a comprehensive assessment of the student's practical and research capabilities, preparing them for further research or professional practice.

BVVS

B.V.V.S

BASAVESHWARENGINEERINGCOLLEGE, BAGALKOTE-587102

Department of Computer Applications (MCA)

For the Admitted Batch 2025-26

MCA-IV Semester Scheme of teaching and examinations for 2026-2027

Sl. No.	Course	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
					Lecture	Tutorial / SDA	Practical	CIE	SEE	Total
1.	PEC/MDC	1PCAA4XXX	(Online Courses)12 weeks duration	3	-	-	-	-	-	100
2.	TS	1PCAA4XXX	Technical Seminar	2	-	-	-	100	-	100
3.	INT	1PCAA4XXX	Research Internship /Industry-Internship / Startup Internship	11	-	-	-	100	100	200
			Total	**16**	-	-	-	200	100	400

INT: Industry/ Research Internship leading to the project work /startup **PROJ:** Project work outcome of Internship (Project Work Phase-II is Viva voce SEE)

TS: Technical Seminar: Students can present the seminar based on the new technologies in the seminar by all postgraduate students of the program shall be mandatory. The CIE marks awarded for the Seminar shall be based on the evaluation of the Report, Presentation skill, and performance in the Question and Answer session in the ratio 50:25:25. Seminar shall be considered as a head of passing and shall be considered for the award of degree. Those, who do not take up/ complete shall be declared as fail in the seminar course and have to complete the same during the subsequent semester.

PCAA401: 1 course Online NPTEL can be cumbersome so one research article may be accepted/Published. A patent can be also one option.

Industry Internship: The main objective of the industry internship is to ensure that the intern is exposed to a real-world environment and gain practical experience. Often, it may be a practical exposure to the theory that has been learned during the academic period. The industry internship helps students understand of analytical concepts and tools, hone their skills in real-life situations, and build confidence in applying the skills learned.

Research Internship: A research internship is an opportunity for students or early career professionals to gain hands-on experience in conducting research under the guidance of a mentor or within a research team. These internships can take place in academic institutions, research organizations, government agencies, or private companies

Research /Industry Internship: In the third-semester Students have to be in touch with a guide/mentor/coordinator and regularly submit the report referred to the progress internship. Based on the progress report the Guide/Mentor/coordinator has to enter the CIE marks. he/she has to attend the SEE at the parent Institute.

MCA I SEMESTER SYLLABUS
(2025-2026)

Course Code: 1PCAA101C	**Programming and Problem Solving in C**	**Credits : 04**
Hours/Week (L:T:P) : 3:0:2		**CIE Marks : 50**
Total Hours of Pedagogy (Theory + Lab): 40 hours Theory + 10 hours Lab		**SEE Marks : 50**
Course Type: Integrated		

Course Objectives:

1. Implement the constructs of C Language.
2. Construct C Programs using basic programming constructs
3. Develop C programs using arrays and strings
4. Organize modular applications in C using functions
5. Integrate pointers and structures in C applications and Execute input/output and file handling in C

UNIT-I	**10Hrs.**

BASICS OF C PROGRAMMING: Problem solving using Algorithm and flowchart, Structure of C program - C programming: Data Types - Constants – Enumeration Constants - Keywords – Operators: Precedence and Associativity - Expressions - Input/ Output statements, Assignment statements Decision making statements - Switch statement. Jumping statements.

UNIT-II	**10Hrs.**

Looping statements – Preprocessor directives - Compilation process. **ARRAYS** Introduction to Arrays: Declaration, Initialization – One dimensional array – Two dimensional arrays **STRINGS:** String operations: length, compare, concatenate, copy –

UNIT-III	**10Hrs.**

FUNCTIONS: Modular programming - Function prototype, function definition, function call, Built-in functions (string functions, math functions) Recursion, Binary Search using recursive functions – **POINTERS:** Pointer operators Pointer arithmetic Arrays and pointers – Array of pointers – Parameter passing: Pass by value, Pass by reference.

UNIT-IV	**10Hrs.**

STRUCTURES: Structure - Nested structures – Pointer and Structures – Array of structures Self referential structures typedef. **Union** - Storage classes and Visibility. **FILE PROCESSING** Files Types of file processing: Sequential access, Random access Sequential access file - Random access file - Command line arguments.

Practical Module

Sl. NO	**Experiments**
1	Simulation of a Simple Calculator.
2	Implement Binary Search on Integers
3	Sort the given set of N numbers using Bubble sort.
4	Implement Matrix multiplication and validate the rules of multiplication.
5	An electricity board charges the following rates for the use of electricity: for the first 200 units 80 paise per unit for the next 100 units 90 paise per unit: beyond 300 units Rs 1 per unit. All users are charged a minimum of R 100 as

	meter charge. If the total amount is more than Rs 400, then an additional surcharge of 15% of total amount is charged. Write a program to read the name of the user, number of units consumed and print out the charges.
6	Write functions to implement string operations such as compare, concatenate, and find string length. Use the functions with parameter passing techniques.
7	Implement structures to read, write and compute average- marks of the students, list the students scoring above and below the average marks for a class of N students.
8	Write a C program to copy a text file to another file, read both the input file name and target file name.
9	Write a C function to find the transpose of a matrix.
10	Write a C program to find the sum of rows and columns of a matrix.

Suggested Learning resources

TEXT BOOKS:

1. Reema Thareja, "Programming in C", Oxford University Press, Second Edition, 2016
2. Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Education, 2015.

REFERENCES:

1. Paul Deitel and Harvey Deitel, "C How to Program with an Introduction to C++", Eighth edition, Pearson Education, 2018.

2. Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020.

3. Byron S. Gottfried, "Schaum's Outline of Theory and Problems of Programming with C", McGraw-Hill Education, 1996.

4. Pradip Dey, Manas Ghosh, "Computer Fundamentals and Programming in C", Second 5. Edition, Oxford University Press, 2013.

5. Anita Goel and Ajay Mittal, "Computer Fundamentals and Programming in C", 1st Edition, Pearson Education, 2013.

Course Outcomes:

CO1: Demonstrate knowledge on C Programming constructs
CO2: Develop simple applications in C using basic constructs
CO3: Design and implement applications using arrays and strings
CO4: Develop and implement modular applications in C using functions
CO5: Develop applications in C using structures and pointers

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**3**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**3**	**-**	**3**	**-**	**-**	**-**	**-**	**-**
CO3	**-**	**-**	**3**	**-**	**-**	**-**	**-**	**-**
CO4	**-**	**-**	**2**	**-**	**-**	**1**	**-**	**-**
CO5	**-**	**-**	**3**	**1**	**-**	**-**	**-**	**-**

Course Code: 1PCAA102C	Mathematical Foundation for Computer Applications	Credits : 03
Hours/Week (L:T:P) : 3:0:0		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50
Course Type: Theory		

UNIT-I	10 Hrs.

Set Theory: Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, Principles of Inclusion and Exclusion, Permutations and Combinations with repetition. **Fundamentals of Logic :**Basic Connectives and Truth Tables, Logic Equivalence- the laws of Logic, Logical Implications, Rules of Inference, The use of Quantifiers, Quantifier Definitions.

UNIT–II	10 Hrs.

Functions: Cartesian products and Relations, Functions-Plain and One-to-One, Onto Functions, Special functions, The Pigeon-hole principle, Function composition and inverse functions. **Relations :**Properties of Relations, Computer recognition-Zero One Matrices and Directed graphs, Posets and Hasse Diagrams

UNIT–III	10 Hrs.

Graph Theory :Graphs and Graphs models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring,

UNIT–IV	10 Hrs.

Statistics: Introduction**,** Measures of central tendency (Arithmetic mean ,Geometric mean, Harmonic mean, Median , Quartiles , Mode). Measures of dispersion (Range, Quartile deviation, mean deviation and standard deviation).**Random variable and probability distribution** Concept of random variable, discrete probability distributions, continuous probability distributions, Mean, variance and co-variance and co-variance of random variables. Binomial and normal distribution, Exponential and normal distribution with mean and variables and problems.

Reference Books *

1. Ralph P. Grimaldi, Discrete and Combinatorial Mathematics an Applied Introduction, 4th Edition, Pearson Education, 2003.
2. Kolman, Busby and Ross, Discrete Mathematics for Computer Science, PHI Publication.
3. A.K.Jaiswal and Anju khandelwal , “Computer based numerical and statistical techniques”, New age international publishers , 2009.

4. Richard A Johnson and C.B Gupta “Probability and statistics for engineers” Pearson Education.
5. H.K.Dass , “Advanced engineering mathematics “, S.Chand Ltd, 2009.
6. Dr. K. S. Chandrashekhar, Engineering Mathematics- IV, Sudha Publications

Course Outcomes

After completion of the course student will be able to

1. Apply the fundamentals of set theory and mathematical logic to perform various set operations and logic to the real world problems.
2. Apply the concept of relations and functions on sets to determine their types and compositions.
3. Acquire ability to work with graph theory.
4. Analyze mathematical concepts like statistics, probability and graph theory to optimize the solutions of engineering problem
5. Demonstrate the ability to use counting techniques, permutations, combinations, and the principles of inclusion and exclusion to solve complex computational and analytical problems.

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**3**	-	-	-	-	-	-	**2**
CO2	**3**	-	-	-	-	-	-	-
CO3	**3**	-	-	**2**	-	-	-	-
CO4	**3**	**2**	**2**	**2**	-	-	-	**3**
CO5	**3**	**2**	**2**	**3**	-	-	-	-

Course Code: 1PCAA103C	Database Management Systems	Credits:03
Hrs/Week:L:T:P:3:0:0		CIE Marks:50
Total Hours/Week:40Hrs		SEE Marks:50

Course Type: Theory

Course Objectives:

1. Analyze the basic concepts and the applications of database systems.
2. Evaluate the different issues involved in the design and implementation of Database System.
3. Explain the basic concepts of relational data model, entity relationship model, relational database design, relational algebra and database language SQL.
4. Design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS

UNIT-I	10Hrs.

Databases and Database Users: Introduction, Characteristics of the Database Approach, Actors on the scene, Workers behind the scene, Advantages of using the DBMS approach, When not to use a DBMS, **Database System Concepts and Architecture:** Data models, schemas and instances, Three-schema architecture and data independence, Database language and interfaces, The database system environment.

Data Modelling Using the Entity-Relationship(ER) Model: Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationships, Relationship Types, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design for the COMPANY Database; ER Diagrams, Naming Conventions and Design Issues.

UNIT–II	10Hrs.

The Relational Data Model and Relational Database Constraint: Relational Model Concepts, Relational Constraints and Relational Database Schemas, Update Operations, Transactions, and Dealing with Constraint Violations. Relational Algebra and Relational Calculus: Unary Relational Operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, Additional Relational Operations, Overview of Tuple Relational Calculus and Domain Relational Calculus; Examples of Queries in Relational Algebra. **Relational Database Design Using ER and EER to-Relational Mapping:** Relational Database Design Using ER to Relational Mapping.

SQL-99: Schema Definition, Constraints, Queries and Views: SQL Data Definition and Data types, Specifying Constraints in SQL, Schema Change statement in SQL, Basic Queries in SQL, Insert, Delete and Update Statements in SQL, Specifying Constraints as Assertion and Triggers, Views in SQL.

UNIT–III	10Hrs.

Functional Dependencies and Normalization for Relational Database: Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form. **Relational Database Design Algorithms and Further Dependencies:** Multi-valued Dependencies and fourth normal form, Join Dependencies and fifth normal form.

UNIT–IV	10Hrs.

Overview of Transaction Management: The ACID Properties: Consistency and Isolation, Atomicity and Durability; Transactions and Schedules; Concurrent Execution of Transactions: Motivation for Concurrent Execution, Serializability, Anomalies due to Interleaved Execution, Schedules Involving Aborted Transactions**; Lock- Based Concurrency Control:** Strict Two-Phase Locking, Deadlocks; Performance of

Locking; Timestamp Based Protocols- Validation- Based Protocols,Multiple Granularity. **Transaction Support in SQL:** Creating and Terminating Transactions, What Should We Lock? Transaction Characteristics in SQL.

Introduction to Crash Recovery: Stealing Frames and Forcing Pages, Recovery - Related Steps during Normal Execution, Overview of ARIES recovery algorithm, Atomicity: Implementing Rollback. Check Points Buffer Management, Failure with loss of nonvolatile storage.

Database Security: Introduction to Database Security; Access Control; Discretionary Access Control: Grant and Revoke on Views and Integrity Constraints; **Mandatory Access Control:** Multilevel Relations and Poly instantiation, Covert Channels, DoD Security Levels.

Suggested Learning resources

Text Books:

1. Elmasri and Navathe, Fundamentals of Database Systems, 6th Edition, Pearson Education, 2011.
2. Raghu Ramakrishnan, JohannesGehrke, Database Management Systems, 3rdEdition,TATA McGrawHill.
3. Silberschatz and Korth, DatabaseSystemConcepts,7th edition, McGrawhill.

ReferenceBooks:

1. C.J.Date, A.Kannan, S.SwamiNadhan, An Introduction to Database systems, 8th Edition, Pearson.
2. M. L.Gillenson, Fundamentals of Database Management Systems, Wiley Student Edition.
3. S.ShahandV.Shah, Oracle for Professionals, The XTeam, SPD.

Course Outcomes:

CO1: Demonstrate the basic elements of a relational database management system.

CO2: Ability to identify and build the data models for relevant problems.

CO3: Design entity relationship and convert entity relationship diagrams into relations and Formulate SQL queries to process the data.

CO4: Ability to analyze the relational model on rules of normal forms.

CO5: Build well-structured transaction.

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**3**	-	-	-	-	-	-	-
CO2	-	**3**	**2**	-	-	-	-	-
CO3	-	**2**	**3**	-	-	-	-	-
CO4	**3**	**2**	-	-	-	-	-	-
CO5	-	-	**3**	**1**	-	-	-	-

Course Code: 1PCAA104C	Operating Systems	Credits : 03
Hours/Week (L:T:P) : 3:0:0		CIE Marks : 50
Total Hours: 40		SEE Marks : 50

Course Type: Theory

Course Objectives:

1. Explain the need and services of the operating system
2. Explore how the operating system handles processes and manages memory.
3. Recognize deadlock condition and technique to handle deadlock situation.
4. Analyze various memory management strategies and file handling concepts.

UNIT-I	10 Hrs.

Introduction to Operating Systems, What operating systems do, Operating System Operations, Computing Environments, Operating Systems Structure, Operating System Services, System Calls, Types of System Calls, System Programs, Operating System Structure, System Boot. Processes, Process Concept, Process Scheduling, Interprocess Communication.

UNIT-II	10 Hrs.

Process Scheduling, Basic Concepts, Scheduling Criteria, Scheduling Algorithms. Synchronization, Background, The Critical Section Problem, Mutex Locks, Semaphores, Classic Problems of Synchronization: Readers-Writers Problem, Dining Philosophers Problem using Semaphores.

UNIT-III	10 Hrs.

Deadlocks, System model, Deadlock Characterization, Methods for handling deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock. Memory Management Strategies Basic Hardware, Swapping, Contiguous Memory Allocation, Segmentation, Paging.

UNIT-IV	10 Hrs.

Virtual Memory Management Background, Demand Paging, Page Replacement. File System, File concept, Access methods, Directory overview Implementing File System Allocation methods, Free Space Management

Suggested Learning resources

Text Books

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Principles, 10th Edition, Wiley India, 2019.

Reference Books:

1. D M Dhamdhere: Operating Systems A Concept Based Approach, 3rd Edition, Tata McGraw Hill, 2017.
2. Harvey M Deital: Operating Systems, 3rdEdition, Addison Wesley, 1990.

Course Outcomes:

CO1: Describe the elements and various functionalities of the operating system
CO2: Apply the techniques of process management and demonstrate process synchronization.
CO3: Recognize deadlock condition and technique to handle deadlock situation.
CO4: Analyze various memory management strategies.
CO5: Describe file handling concepts.

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**1**	**2**	**3**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**1**	**3**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**3**	**1**	**-**	**-**	**-**	**-**	**-**
CO4	**3**	**2**	**1**	**-**	**-**	**-**	**-**	**-**
CO5	**1**	**2**	**3**	**-**	**-**	**-**	**-**	

Course Code: 1PCAA105C	Computer Networks	Credits : 03
Hours/Week (L:T:P) : 3:0:0		CIE Marks : 50
Total Hours: 40		SEE Marks : 50

Course Type: Theory

Course Objectives:

1. Understand computer networks fundamentals and issues.
2. Appreciate computer network models and role of layers in each model.
3. Understand different protocols and techniques supported at each level of network software
4. Comprehend application and challenges of computer networks.

UNIT-I	10 Hrs.

Introduction: Uses of Computer Networks, Network Hardware. Network Software: Protocol Hierarchies, Design Issues for the Layers. Types of Networks, Topologies, **Reference Models:** The OSI Reference Model, The TCP/IP Reference Model, A Comparison of the OSI and TCP/IP Reference Models. Physical Layer- Guided Transmission Media, Digital Modulation and Multiplexing.

UNIT-II	10 Hrs.

Data Link Layer-Data link Layer Design issues, Framing, Flow Control and Error Correcting and Detection codes, Sliding Window Protocols (Stop and Wait, Go-Back-N (GBN) and Selective Repetitive (SR)), Medium Access Control-The Channel Allocation Problem, Multiple Access Protocols.

Data Link Layer Switching: Uses of bridges, repeaters, hubs, switches, routers and gateways.

UNIT-III	10 Hrs.

The Network Layer: Network Layer Design issues, Routing algorithms- The Optimality Principal, Shortest Path Algorithm, Flooding, Distance Vector Routing, Link State Routing, Hierarchical routing, Congestion Control Algorithms, Quality of Service, Internetworking. The Network Layer in the Internet: The Network Layer in the internet- IP version 4 Protocol(IPv4), The Main IPv6 Header, Extension Headers,

Internet Control Protocols: ICMP, ARP, DHCP.

UNIT-IV	10 Hrs.

The Transport Layer -The Transport Service: Services Provided to the Upper Layers, Berkeley Sockets, Elements of Transport Protocols, Internet transport protocols.TCP: Introduction to TCP, The Service Model, Protocol, Segment Header, UDP.

The Application Layer- The Domain Name System, Electronic Mail, The World-Wide-Web.

Suggested Learning resources

Text Books

1. Andrew S. Tanenbaum, David J Wetherall, “Computer Networks”, Pearson Education, Pearson Publication, 5th Edition, 2012.
2. Behrouz A Forouzan, Firouz Mosharraf, “Computer Networks A Top-Down Approach”, Tata McGraw-Hill Education Pvt. Ltd, 2011.
3. William Stallings, "Data and Computer Communication" ,8th edition, Pearson Publications,2007.

Course Outcomes:

After completion of the course student will be able to

CO1: To comprehend basics of data communication system.

CO2: Enumerate the layers of the OSI, TCP/IP model and demonstrate functions of each layer and comprehend the concept of data link protocols.

CO3: To exhibit the ability to apply different error detection and correction technique to solve communication problem.

CO4: To exhibit the ability to understand issues related to transport layer and protocols.
CO5: Demonstrate the concept of internetworking, routing techniques of network layer.

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**2**	**1**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**1**	**1**	**-**	**-**	**-**	**-**	**-**
CO4	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**
CO5	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**

Course Code: 1PCAA106L	Database Management System Laboratory	Credits:2
Hours/Week (L:T:P) : 0:0:4		CIEMarks:50
Total Hours of Pedagogy (Theory + Lab):		SEEMarks:50

1. Create the following tables with properly specifying Primary keys, Foreign keys for the medical consultation center.

   SPECIALIZATION(Sp_Id, Sp_Name, Sp_Description)

   Doctor (D_Id,D_Name, Sp_Id)

   PATIENT (P_Id, P_name, p_address, date_of_ registration)

   CONSULTATION DATE (C_date, D_Id, P_Id).

   Write the SQL statements for the following queries.

   i. Generate list of patients and their consultation detail.

   ii. Find patients consulted by specific doctor.

   iii. List the patients by the their date of registered

   iv. Find the details of the entire faculty based on specialization and have consultation date next month.

   v. Find the patient whose consultation date is today along with the concerned faculty detail.

2. Create the following tables with properly specifying Primary keys, Foreign keys for the library system.

   Faculty(ID, Name, Designation, DoB, Dept_ID)

   BRANCH (ID, Name, HoD)

   STUDENT (USN, Name, Address, BranchId, Sem)

   BOOK (BookId, Bookname, Authorid, Publisher, BranchId)

   AUTHOR (Authorid, Authorname, Country,DoB)

   BORROW (USN, Bookid, Borrowed_Date)

   Write the SQL statements for the following queries.

   i. List the details of Students who are all studying in 2nd sem MCA.

   ii. Display the Book names in descending order of their names.

   iii. List the names of HoDs of all the departments.

   iv. List all the books by author, branch, publisher etc.

   v. Display the faculty details in descending order of their experience.

3. Create the following tables with properly specifying Primary keys, Foreign keys for the vehicle insurance company.

   PERSON (Driver_Id, D_Name, address)

   CAR (Reg_No, Model, Year_of_Manunfacture, Make)

ACCIDENT (Report_No, Adate, Location)

OWNS (Driver_id, Reg_No)

PARTICIPATED (Driver_id, Reg_No, Report_No, Damage_Amount)

Write the SQL statements for the following queries.

i. Update the damage amount for the car with a specific registration number in the accident with a given report number to a specified amount.

ii. Find the owner of car that is involved in the accident of specific date.

iii. Find the total damage amount out of accidents between given dates.

iv. Find the car that is frequently meeting an accident.

v. Find more than one car frequently meeting the accident belongs to a owner.

4. Create the following tables with properly specifying Primary keys, Foreign keys for the bank system.

Customer(Cust_Id,Cust_Name,Address,DoB,PAN)

Branch(Branch_Id,Branch_Name,Location)

Account(Account_No,Account_Type,Balance,Branch_Id)

Transaction(Trans_Id,Account_No,Date_of_Trans,Trans_Type,Amount)

Opens_Account(Cust_Id,Account_No,Date_of_Creation)

Write the SQL statements for the following queries.

i. List the details of account holders who have a both accounts in the same branch.

ii. List the Name and address of account holders who have more transactions in current month.

iii. List the total number of transactions, amount deposited and amount withdrawn of a customer with given a Permanent Account Number (PAN) of a specific financial year.

iv. Display the total amount deposited and withdrawn of a branch in a specific month.

v. Display the date on which the maximum number of withdrawal transactions occurred in a specific branch.

5. Design an ER-diagram for the following scenario, Convert the same into a relational model.

Consider a Cricket Tournament “ABC CUP” organized by an organization. In the tournament many teams are contesting, each having a TeamId,Team_Name, City and a coach. Each team is uniquely identified by TeamId. A team can have many Players and a captain. Each player is uniquely identified by PlayerId, having a Name, and multiple phone numbers andDoB. A player represents only one team but can play many matches. There are many Stadiums to conduct matches and each stadium is identified by StadiumId, having a stadium_name, Address (involves city, area_name, pincode). A team can play many matches andeach match played between the two teams in the scheduled date and time in the predefined Stadium. Each match is

identified uniquely by using MatchId and match won by any one team and that also recorded in the database. For each match man_of_the match award given to a player.

Write the SQL statements for the following queries.

i. Display the youngest player (in terms of age) Name, Team name , age

ii. List the details of the stadium where the maximum number of matches were played.

iii. List the details of the player who is not a captain but got the man_of _match award at least in two matches.

iv. Display the Team details who won the maximum matches.

v. Display the team name where all its won matches played in the same stadium.

6. Referring problem definition of question 1, write the SQL statements for the following queries.

i. Find the doctors who have no consultations between dates.

ii. Generate up-to date list of doctor-wise the number of consultations.

iii. Generate list of patients who visited the center for consultation at least three times.

iv. Find the doctors who have maximum consultations.

v. Find the patient who is consulted by all faculties.

7. Referring problem definition of question 3, write the SQL statements for the following queries.

i. Find the total number of people who owned cars that were involved in accident in 2024.

ii. Find the number of accidents in which cars belonging to a specific model were involved.

iii. Find the model highly involved in the accidents.

iv. Find the vehicle that is involved in the accident at sensitive location.

v. Find the driver who is involved in more number of accidents at sensitive location.

8. Referring problem definition of question 4, write the SQL statements for the following queries.

i. Count the number of accounts of different account type in all branches of the same city.

ii. Display all the joint accounts with number of customers.

iii. Find all customers who have an account at all the branches located in 'Bagalkote'.

iv. Display the total amount of all the customers of all the branches located in ‘Bagalkote’.

v. Add 6% interest on savings accounts of senior citizens and 4.5% on all other savings accounts.

9. Create the following tables with properly specifying Primary keys, Foreign keys for an

order processing company.

CUSTOMER (Cust_Id, Cust_Name, City)

ITEM (Item_Id, Unit_Price)

WAREHOUSE (Warehouse_No, City)

CUST_ORDER (Order_No, Cust_Id ,Ord_Date, Ord_Amt)

ORDER-ITEM (Order_No, Item_Id, Qty)

SHIPMENT (Order_No, Warehouse_No, Ship_Date)

Write the SQL statements for the following queries.

i. Display detail of orders of specific customer.

ii. Display detail of orders of the items in a proper order.

iii. List the customer who has purchased a specific item more number of times between dates.

iv. Produce a listing: CustName, Orders, Avg_Order_Amt, where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer.

v. List the orders that were shipped from all the warehouses that the company has in a specific city.

10. Referring problem definition of question 9, write the SQL statements for the following queries.

i. Demonstrate how you delete an Item with a given number from ITEM table and make that field null in the ORDER-ITEM table.

ii. Generate the list of orders to be shipped next month.

iii. Display the list of orders whose customer exists in the same city that of the warehouse.

iv. To display which item is highly demanded by the customers.

v. Display the customer who purchased the items together frequently.

Course Outcomes (COs):

After completion of the course student will be able to:

1. Create database objects like table, constraint, view and index.
2. Design entity-relationship diagrams to solve given database applications.
3. Formulate SQL queries for the given problem.
4. Design PL/SQL like stored procedure, trigger etc..

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**3**	**1**	**1**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**-**	**3**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**3**	**-**	**-**	**-**	**-**	**-**	**-**
CO4	**2**	**-**	**3**	**1**	**-**	**-**	**-**	**-**
CO5	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**

Course Code: 1PCAA107L	Web Technologies Laboratory	Credits : 02
Hours/Week (L:T:P) : 0:0:2		CIE Marks : 50
Total Hours of Pedagogy (Theory + Lab):		SEE Marks : 50

Course Type: Practical

Course Objectives:

1. Create database objects that include tables, constraints, Views and indexes.
2. Create SQL queries to read information form tables.
3. Design Web page.
4. Building interactive web pages.

Sl. No.	Experiments
01	Design and develop a static web page using basic HTML tags to demonstrate use of different color, font, table format, bold, italic etc.
02	Design and develop a web page to demonstrate different types of style sheets.
03	Design and develop external CSS style sheet to create a login page form and validate using JavaScript.
04	Write a JavaScript to demonstrate use of alert, prompt and confirm message box by considering general feedback form.
05	Develop and demonstrate a XHTML file that includes Javascript script for the following problems: a) Input : A number n obtained using prompt Output : The first n Fibonacci numbers
06	Demonstrate use of hyperlink to pass parameters and validate those parameters in second page using JavaScript
07	Design and develop a Registration form Forward the request to success and failure page by validating user credentials using JavaScript.
08	Create a HTML form containing textbox to enter text. Write a JavaScript code block, which checks the contents entered in a form's Text element. If the text entered is in the lower case, convert to upper case. Make use of function to Uppercase( ).
09	a) Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page. b) write PHP program to store current date-time in a COOKIE and display the "Last visited on" date –time on the web page upon reopening of the same page.
10	Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text

fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.

Course Outcomes (COs):

CO1: Create basic static web pages using HTML tags and formatting.
CO2: Apply different CSS styles to enhance webpage design.
CO3: Use JavaScript for interactivity and form validation.
CO4: Implement PHP programs using sessions and cookies.
CO5: Store and retrieve form data using PHP and MySQL.

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	2	3	4	5	6	7	8
CO1	3	-	2	-	-	-	-	-
CO2	3	-	2	3	-	-	-	-
CO3	-	-	2	2	-	-	-	-
CO4	-	-	-	3	-	-	-	-
CO5	-	-	-	-	-	-	-	-

MCA II SEMESTER SYLLABUS
(2025-2026)

1PCAA201C	Introduction to Python	Credits:04
Hours/Week (L:T:P) : 3:0:2		**CIEMarks:50**
Total Hours of Pedagogy (Theory + Lab): 40 hours Theory + 10 hours Lab **Course Type: Integrated**		**SEEMarks:50**

Hours/Week (L:T:P) : 3:0:2

Course Objectives:

1. Learn fundamental features of object-oriented language and JAVA
2. Set up Java JDK environment to create, debug and run simple Java programs.
3. Implement core language constructs, including data types, operators, control statements, arrays, And exception handling mechanisms, with examples.
4. Study the concepts of importing of packages and exception handling mechanism.
5. Utilize the Java Collections Framework and understand how to develop modular, well-documented Code.

UNIT-I	10 Hrs.

Rapid Introduction to Procedural Programming: Creating and Running Python Programs; Python's "Beautiful" Objects; Data Types (Object References, Collection Types); Control Structures and Functions. Data Types: Integral Types (Integers, Booleans); Floating-Point Types (Floating-Point Numbers, Complex Numbers, Decimal Numbers); Strings (Comparing, Slicing, Formatting). Collection Types: Tuple and Named Tuples; Lists (Comprehensions); Sets (Sets, Frozensets); Mappings (Dictionaries, Dictionary Comprehensions).

UNIT-II	10 Hrs.

Control Structures and Functions: Control Structures (Conditional Branching, Looping); Exception Handling (Catching, Raising); Custom Functions (Namespaces, Arguments, Lambda Functions, Assertions). Modules: Custom Modules and Packages; Overview of Python's Standard Library (String Handling, Command-Line Programming, Mathematics, Dates and Times, Algorithms and Collections, File Formats, File System, Networking).

UNIT-III	10 Hrs.

Object-Oriented Programming: The Object-Oriented Approach; Custom Classes (Attributes and Methods, Inheritance and Polymorphism, Using Properties to Control Attribute Access); Creating Complete Fully Integrated Data Types. File Handling: Writing and Reading Binary Data (Pickles, Raw Binary Data); Writing and Parsing Text Files (CSV, JSON); Writing and Parsing XML Files (ElementTree, DOM, SAX).

UNIT-IV	10 Hrs.

Advanced Programming Techniques: Advanced Function and Module Features; Advanced Attribute Access; Metaclasses; Functional Programming Techniques. Processes and Threading: Introduction to the Multiprocessing Module; Introduction to the Threading Module; Networking (TCP/IP Client/Server). Regular Expressions: The re Module; Using Regular Expressions to Match, Search, and Replace Text.

Suggested Learning resources

1. Summerfield, M., "Programming in Python 3: A Complete Introduction to the Python Language", Addison-Wesley Professional, 2nd Edition, 2009.

Reference Books*

1. Downey, A. B., "Think Python: How to Think Like a Computer Scientist", O'Reilly Media, 2nd Edition, 2016.
2. Lutz, M., "Learning Python", O'Reilly Media, 5th Edition, 2013.
3. Brown, M. C., "Python: The Complete Reference", McGraw-Hill Education, 4th Edition, 2018.

Course Outcomes:

After completion of the course student will be able to

CO1: Analyze and utilize Python's diverse built-in data types and collection structures for efficient data management and manipulation.

CO2: Design and implement modular, reusable code using custom functions, namespaces, and advanced modules.

CO3: Apply object-oriented principles, including inheritance and polymorphism, to create fully integrated and custom data types.

CO4: Develop robust applications capable of processing diverse file formats, including binary data, CSV, JSON, and XML.

CO5: Implement high-level concurrency and network communication using threading, multi-processing, and socket programming.

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**2**	**1**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**1**	**1**	**-**	**-**	**-**	**-**	**-**
CO4	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**
CO5	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**

1PCAA202C	Object Oriented Programming using JAVA	Credits:04
Hrs/Week: L:T:P:S 3:0:3:0		CIEMarks:50
Total Hours/Week: 52Hrs		SEEMarks:50

Course Type: Theory

Course Objectives:

1. Learn fundamental features of object-oriented language and JAVA
2. Set up Java JDK environment to create, debug and run simple Java programs.
3. Implement core language constructs, including data types, operators, control statements, arrays, And exception handling mechanisms, with examples.
4. Study the concepts of importing of packages and exception handling mechanism.
5. Utilize the Java Collections Framework and understand how to develop modular, well-documented Code.

UNIT-I — 10 Hrs.

Overview of Java Basics:

The History and Evolution of Java - Overview of Java – Program Structure- Object-Oriented Programming, A First Simple Program, A Second Short Program, Two Control Statements, Using Blocks of Code, Lexical Issues, Java Keywords, The Java Class Libraries -Data Types, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, A Closer Look at Literals, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays : One dimensional Arrays, Multi-dimensional Arrays,

Operators : Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses -Control Statements- Java's Selection Statements, Iteration Statements, Jump Statements.

Text book 1: Ch 1, Ch 2, Ch 3, Ch 4, Ch 5

Teaching Learning Process : Chalk &Talk , Power Point Presentation , RBT levels L1, L2, L3

UNIT-II — 10 Hrs.

Introducing Classes, Closer look at Methods and Classes, INHERITANCE, PACKAGES AND INTERFACE:

Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize( ) Method, A Stack Class, A Closer **Look at Methods and Classes:** Overloading Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static, Introducing final, Arrays Revisited, Exploring String Class , Using Command Line Arguments, Var args.

Inheritance: Member Access and Inheritance – Multilevel Hierarchy - Method Overriding – Dynamic Method Dispatch - Keywords: Abstract – Super - Final –The Object Class

Packages: Defining a Package - Access Protection - Importing Packages. Interfaces: Defining an Interface - Implementing Interfaces, Nested Interfaces, Variables and Methods in Interface.

Text book 1: Ch 6, Ch 7, Ch 8, Ch 9

Teaching Learning Process : Chalk &Talk , Power Point Presentation , RBT levels L1, L2, L3.

UNIT-III	10 Hrs.

Exception handling, Multithreading, Enumerations : Exception handling: Exception Types – Try and Catch statement – Throw – Throws – Finally. Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions.

Multithreading: The Java Thread Model – Creating Threads and Multiple Thread Creation - Thread Priorities – Synchronization – Deadlock –Obtaining a Thread's State

Enumerations, Auto boxing and Annotations – Enumerations Fundamentals, Type Wrappers, Autoboxing and Methods, Annotations Basics

Text book 1: Ch 9, Ch 10, Ch 11, Ch 12

Teaching Learning Process : Chalk &Talk , Power Point Presentation , RBT levels L1, L2, L3

UNIT-IV	10 Hrs.

I/O Packages: I/O basics, Byte Streams -Character Streams – Reading console input - Writing Console output, Print writer class , Reading and Writing Files, Native methods, Static Import.

Collections Framework: Collection Interfaces: Set - Sorted Set – List – Queue - Dequeue – Collection Classes: Array List -Linked List- Hash Set – Priority Queue – Comparators.

Generics – Generics Classes, General form of Generic Class, Generic Interfaces, Generic Class Hierarchies.

String Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf( ), Changing the Case of Characters Within a String , Additional String Methods, String Buffer, String Builder.

Text book 1: Ch 13, Ch 14, Ch 18

Teaching Learning Process : Chalk &Talk , Power Point Presentation , RBT levels L1, L2, L3

Suggested Learning resources

Text Books

1. Herbert Schildt, Java The Complete Reference, 13th Edition, Tata McGraw Hill, 2024. (Chapters 2, 3, 4, 5, 6,7, 8, 9,10, 12,13,15)

Reference Books*

1. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming withjava, Tata McGraw Hill education private limited.
2. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies.
3. Anita Seth and B L Juneja, JAVA One step Ahead, Oxford University Press, 2017.

Course Outcomes:

After completion of the course student will be able to

CO1: Understand the fundamental principles of Java and the Object-Oriented Programming (OOP) paradigm.

CO2: Apply concepts of classes, objects, and encapsulation to write functional programs.

CO3: Implement code reusability and modularity using inheritance, interfaces, and packages.

CO4: Develop robust applications by handling errors through exception handling and multithreading.

CO5: Integrate the concepts of event handling, components and controls for developingapplications

CO-PO Mapping

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**1**	**2**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**2**	**2**	**1**	**-**	**-**	**-**	**-**	**-**
CO3	**2**	**1**	**1**	**-**	**-**	**-**	**-**	**-**
CO4	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**
CO5	**-**	**1**	**2**	**-**	**-**	**-**	**-**	**-**

Course Code: 1PCAA203C	Data Structures and Algorithms	Credits : 04
Hours/Week (L:T:P) : 4:0:0		CIE Marks : 50
Total Hours: 52		SEE Marks : 50

Course Type: Theory

Course Objectives:

1. To understand and implement fundamental data structures.
2. To develop efficient algorithms for solving problems.
3. To analyze the time and space complexity of algorithms.
4. To gain practical experience in applying data structures and algorithms to real-world problems.
5. To prepare students for industry roles requiring strong foundations in data structures and algorithmic thinking.

UNIT I	13 Hrs.

Introduction to Data Structures and Algorithms: Basic Concepts: Definition and importance of data structures, Abstract Data Types (ADTs), Algorithm analysis: Time and space complexity, Big O notation. **Arrays:** Definition and operations: Insertion, deletion, traversal, Multidimensional arrays, Applications of arrays. **Linked Lists:** Singly linked list: Creation, insertion, deletion, traversal, Doubly linked list and circular linked list, Applications of linked lists.

UNIT II	13 Hrs.

Stacks and Queues,: Stacks: Definition and operations: Push, pop, peek, Applications: Expression evaluation, backtracking, function calls. **Queues:** Definition and operations: Enqueue, dequeue, front, rear, Types: Circular queue, priority queue, double-ended queue (deque), Applications of queues. **Recursion:** Definition and principles of recursion, Recursive algorithms: Factorial, Fibonacci series, Tower of Hanoi, Analysis of recursive algorithms.

UNIT III	13 Hrs.

Trees: Definition and terminology: Root, leaf, internal node, height, depth, Binary trees: Traversal (preorder, inorder, postorder), creation, insertion, deletion, Binary search trees (BST), AVL trees, B-trees. **Graphs:** Definition and terminology: Vertices, edges, adjacency list, adjacency matrix, Graph traversal algorithms: Depth-first search (DFS), breadth-first search(BFS), Shortest path Algorithms: Dijkstra's algorithm Floyd - Warshall algorithm.

UNIT IV	13 Hrs.

Sorting and Searching Algorithms: Sorting Algorithms: Basic concepts and classification, Comparison-based sorting: Bubble sort, selection sort, insertion sort, quicksort, mergesort, heapsort, Non-comparison-based sorting: Radix sort, counting sort. **Searching Algorithms:** Linear search and binary search, Search in linked lists, trees, and graphs, Hashing: Hash functions, collision resolution techniques (chaining, open addressing). **Advanced Data Structures and Applications:** Advanced Data Structures: Heaps: Definition, operations, heap sort, applications, Trie: Definition, operations, applications in dictionary and spell-checking,

Suggested Learning resources

Text Books:

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2023.
2. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
3. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

Reference books:

1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning,2014.
2. A M Tenenbaum, Data Structures using C, PHI, 1989
3. Introduction to the Design and Analysis of Algorithms, Anany Levitin: 2nd Edition, 2009. Pearson.
4. Algorithms, Kenneth A Berman and Jerome L Paul, Cengage Learning India Pvt Ltd, 2002 edition.

Course Outcomes:

CO1: Demonstrate an understanding of fundamental data structures and algorithms
CO2: Implement and manipulate data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
CO3: Develop algorithms for searching, sorting, and optimization problems.
CO4: Analyze the efficiency and correctness of algorithms.
CO5: Apply data structures and algorithms to solve complex problems in various domains.

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	**X**	**X**						
CO2			**X**					
CO3				**X**				
CO4								**X**
CO5								**X**

1PCAA204C	Software Engineering	Credits: 03
Hrs/Week:L:T:P:S **3:0:0:0**		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50

Course Objectives:

1. To understand the fundamental principles, processes, and ethics of software engineering.
2. To apply requirements engineering, system modeling, and object-oriented design techniques in software development.
3. To develop skills in software testing, quality management, and project planning.
4. To understand Agile methodologies and Scrum practices for managing software projects.
5. To explore modern software development practices including continuous integration and continuous deployment.

UNIT-I	10 Hrs.

Software Process & Software Development Methods: Professional Software Development, software engineering ethics, Software process models, Process activities, Coping with change. Agile Software Development: Agile methods, Plan-driven and agile development, Extreme programming, Requirements Engineering: Functional and non-functional requirements, The software requirements document, Requirements specification. Requirements engineering processes, Requirements elicitation and analysis, Requirements validation. Requirements change.

UNIT–II	10 Hrs.

System Models: Context models, Interaction models, Structural models, Behavioral models, Model-driven engineering, Design and Implementation: Introduction to RUP, Design Principles. Object-oriented design using the UML Design patterns, Implementation issues, Open source Development.

UNIT–III	10 Hrs.

Software testing: Development testing, Test-driven development, Release testing, User testing. Project management: Risk management, Managing people, Teamwork, Project planning, Software pricing, Plan-driven development, Project scheduling, Agile planning, and Estimation techniques. **Quality management** : Software quality, Software standards, Reviews and Inspections, Software measurement and metrics.

UNIT–IV	10 Hrs.

Scrum Framework Foundation of Scrum, pillars of empiricism, Scrum Values, Identifying a Scrum Team; Scrum Events: Spring planning, Implementation and review, Scrum Artifacts: Creating, Managing and refining product backlog. Scrum in Action: Planning and Estimating with Scrum: Estimation Scale, Bucket method, Envisioning the product journey with a product roadmap; Sprint Journey: Refining the Product Backlog, Tracking progress with a Scrum Board, Defects in Sprint; Facets of Scrum: software development practices for Scrum, Source control model for continuous integration, Continuous delivery and continuous deployment, Leveraging testing methods for Scrum, Applying Scrum to remote teams.

Reference Books:

1. Ian Sommer ville, Software Engineering, 10th Edition, Pearson India Education Services Pvt. Ltd., 2020.
2. Roger S. Pressman, Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
3. Pankaj Jalote, An Integrated Approach to Software Engineering, Wiley India.
4. Fred heath, The Professional Scrum Master Guide, Packt Publishing, 2021.
5. Stacia Viscardi, The Professional Scrum Master's Handbook, Packt Publishing, 2013
6. Andrew T. Pham and David K. Pham, Business-Driven IT-Wide Agile (Scrum) and Kanban (Lean) Implementation, CRC Press.

Course Outcomes:

1. Understand software process models, agile methods, and requirements engineering concepts.
2. Identify and analyze functional and non-functional requirements for software systems.
3. Develop system models and software designs using UML and object-oriented principles.
4. Apply software testing, project management, and quality management techniques.
5. Understand and apply Scrum framework and agile practices in software development.

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	2	2	1	2	-	-	1	1
CO2	1	3	2	-	-	-	-	1
CO3	1	2	3	2	1	-	-	1
CO4	1	2	2	3	2	3	1	1
CO5	-	1	2	2	3	2	1	2

1PCAA205C	Introduction to AI	Credits: 03
L:T:P - 3:0:0		CIEMarks:50
Total Hours/Week: 40 Hrs		SEEMarks:50
Course Type: Theory		

Course Objectives:

1. To understand the fundamentals, types, and applications of Artificial Intelligence.
2. To learn intelligent agents and knowledge representation techniques for AI systems.
3. To develop effective prompt engineering skills for interacting with Large Language Models.
4. To apply search and optimization techniques for AI-based problem solving.
5. To explore Machine Learning concepts, AI trends, ethics, and industrial applications.

UNIT-I	10 Hrs.

Introduction to Artificial Intelligence: Artificial Intelligence, How Does AI Work?, Advantages and Disadvantages of Artificial Intelligence, History of Artificial Intelligence, Types of Artificial Intelligence, Weak AI, Strong AI, Reactive Machines, Limited Memory, Theory of Mind, Self-Awareness, Is Artificial Intelligence Same as Augmented Intelligence and Cognitive Computing, Machine Learning and Deep Learning.

Machine Intelligence: Defining Intelligence, Components of Intelligence, Differences Between Human and Machine Intelligence, Agent and Environment, Types of an agent.

Knowledge Representation: Introduction, Knowledge Representation, Knowledge-Based Agent, Types of Knowledge.

UNIT–II	10 Hrs.

Prompt Engineering: Introduction to Prompt Engineering, The Evolution of Prompt Engineering, Types of Prompts, How Does Prompt Engineering Work?, Comprehending Prompt Engineering's Function in Communication, The Advantages of Prompt Engineering, The Future of LLM Communication.

Prompt Engineering Techniques for ChatGPT, Introduction to Prompt Engineering Techniques, Instructions Prompt Technique, Zero, One, and Few Shot Prompting, Self-Consistency Prompt.

UNIT–III	10 Hrs.

Search Techniques: Solving Problems by Searching, Problem-Solving Agents, Searching for Solutions. **Uninformed Search Strategies:** Breadth-First Search, Depth-First Search, Depth-Limited Search, Bidirectional Search, and Comparing Uninformed Search Strategies. **Heuristic (Informed) Search Strategies:** Greedy Best-First Search, A* Search, Memory-Bounded Heuristic Search. **Local Search Algorithms and Optimization Problems**: Hill-Climbing Search, Simulated Annealing, Local Beam Search, Genetic Algorithms**. Constraint Satisfaction Problems**: Constraint Satisfaction Problems and Local Search for Constraint Satisfaction Problems.

UNIT–IV	10 Hrs.

Machine Learning: Techniques in AI, Machine Learning Model.

Trends in AI: AI and Ethical Concerns, AI as a Service (AIaaS).

Applications of AI: Robotics, Robotics-an Application of AI, Drones Using AI, No Code AI, Low Code AI.

Industrial Applications of AI: Application of AI in Healthcare, Application of AI in Finance,

Application of AI in Retail, Application of AI in Agriculture, Application of AI in Education, Application of AI in Transportation, AI in Experimentation and Multi-disciplinary research.

Text Books

1. Reema Thareja, (2023), Artificial Intelligence: Beyond Classical AI, Pearson Education.
2. Ajantha Devi Vairamani and Anand Nayyar, (2024), Prompt Engineering: Empowering Communication, (1st Edition), CRC Press, Taylor & Francis Group.
3. Saptarsi Goswami, Amit Kumar Das and Amlan Chakrabarti, (2024), AI for Everyone – A Beginner's Handbook for Artificial Intelligence, Pearson Education.

Reference Books

1. Stuart Russell and Peter Norvig, (2023), Artificial Intelligence: A Modern Approach (4th Edition), Pearson Education.
2. Elaine Rich, Kevin Knight, and Shivashankar B. Nair, Artificial Intelligence, McGraw Hill Education.
3. Tom Taulli, Prompt Engineering for Generative AI: ChatGPT, LLMs, and Beyond, Apress, Springer Nature.
4. Nilakshi Jain, Artificial Intelligence: Making A System Intelligent, (First Edition), Wiley.

Course Outcomes

After completion of the course student will be able to

1. Explain the concepts and types of artificial intelligence.
2. Illustrate basic machine learning methods for regression, classification and clustering.
3. Identify real-world applications across different disciplines.
4. Make use of prompt engineering techniques to interact with generative AI tools.
5. Outline recent trends in artificial intelligence and machine learning.

Course Outcomes	Programme Outcomes							
	1	2	3	4	5	6	7	8
CO1	3	-	2	3	1	-	1	1
CO2	3	3	3	3	1	-	1	2
CO3	3	3	3	3	1	-	2	3
CO4	3	3	3	3	2	-	3	3
CO5	2	3	2	3	1	-	3	3

Course Code: 1PCAA206L	Data Structures with Algorithms Laboratory	Credits : 02
Hours/Week (L:T:P) : 0:2:0		CIE Marks : 50
Total Hours of Pedagogy : 30		SEE Marks : 50
Course Type: Practical		

Course Objectives:

1. Evaluate the Expressions like postfix, prefix conversions.
2. Implementing various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs.

Sl. No.	Experiments
01	Write a well structured C program to insert, delete and display elements in an array depending on user choice.
02	Write a well structured C program to implement different operations of a stack with array.
03	Implement a Program in C for converting an Infix Expression to Postfix Expression.
04	Design, develop, and execute a program in C to evaluate a valid postfix expression using stack. Assume that the postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), * (multiply) and / (divide)
05	Design, develop, and execute a program in C to simulate the working of a queue of integers using an array. Provide the following operations: a. Insert b. Delete c. Display
06	Write a C program to simulate the working of a singly linked list providing the following operations: a. Display & Insert b. Delete from the beginning/end c. Delete a given element
07	Write a C program to Implement the following searching techniques a. Linear Search b. Binary Search.
08	Write a C program to implement the following sorting algorithms using user defined functions: a. Bubble sort (Ascending order) b. Selection sort (Descending order).
09	Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm (C programming)
10	From a given vertex in a weighted connected graph, find shortest paths to other vertices Using Dijkstra's algorithm (C programming)

Course Code: 1PCAA207L	Object Oriented Programming using JAVA LAB	Credits : 02
Hours/Week (L:T:P) : 0:2:0		CIE Marks : 50
Total Hours of Pedagogy : 30		SEE Marks : 50
Course Type: Practical		

Course Objectives:

1. To provide hands-on experience in developing Java programs using object-oriented programming concepts.
2. To implement fundamental programming constructs such as classes, objects, constructors, methods, arrays, strings, and exception handling.
3. To develop practical skills in using Java collections, interfaces, inheritance, and file handling mechanisms.
4. To enable students to design, implement, test, and debug Java applications for solving real-world problems.

Sl. No.	Experiments
01	Write a JAVA program to implement a Stack Program.
02	Write a JAVA program to enter a number from keyboard find out factorial of the number.
03	Write a JAVA program to demonstrate Constructor Overloading and Method Overloading.
04	Write a JAVA program for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
05	Write a JAVA program to show use of throw, throws and finallykeyword.
06	Write a JAVA program to implement a Class called Account that creates account with Rs. 500 minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than Rs. 500.
07	Write a JAVA program to find out the biggest and smallest number from a matrix.
08	Write a JAVA program to enter a string from keyboard and check how many vowels and consonants are present.
09	Write a JAVA program to implement HashMap example to add(), remove() and replace() elements.

10	Write a JAVA program which uses FileInputStream / FileOutPutStream Classes.

Outcomes (Course COs)

CO1: Apply Java programming fundamentals, constructors, methods, arrays, and string manipulation techniques to develop simple Java applications.

CO2: Implement object-oriented programming concepts such as inheritance, interfaces, constructor overloading, and method overloading in Java programs.

CO3: Develop Java applications using exception handling mechanisms and collection framework classes such as HashMap.

CO4: Design and implement Java programs involving file handling, matrix operations, and problem-solving techniques to develop robust applications.

CO5: Analyze, test, and debug Java programs to ensure correctness, reliability, and efficient execution.

MCA III SEMESTER SYLLABUS

(2025-26)

Course Code: 1PCAB301E	Data Mining and Data Warehousing	Credits : 03
Hours/Week (L:T:P) :3:0:0		CIE Marks : 40
Total Hours: 03		SEE Marks : 50

Corse Objectives:

1. To gain knowledge on data mining and the need for pre-processing.
2. To characterize the kinds of patterns that can be discovered by association rule mining.
3. To get exposed to the concepts of data warehousing architecture and implementation.
4. To implement classification techniques on large datasets.
5. To analyze various clustering techniques in real world applications.

UNIT-I	10 Hrs.

Introduction and Data Pre-processing :Why data mining, What is data mining, What kinds of data can be mined, What kinds of patterns can be mined, Which Technologies Are used, Which kinds of Applications are targeted, Major issues in data mining .DBMS VS Data Mining. Data Pre-processing: An overview, Data cleaning, Data integration, Data reduction, Data transformation and data discretization.

UNIT -II	10 Hrs.

Need for Data Warehouse: basic model, characteristics, Data Warehouse Architecture, Database versus Data Warehouse Multidimensional Data Model– Schemas for Multidimensional Databases– OLAP operations, types of data warehouse, Data Warehouse components. OLAP versus OLTP, characteristics of OLAP, OLAP types.

UNIT -III	10 Hrs.

Association Analysis: Definition of Association Analysis, Frequent Item Set Generation, Rule Generation, Compact Representation of Frequent Item Sets. Alternate Method of Generating Item Sets, FP Growth Algorithms, Evaluation of Association Pattern.

UNIT -IV	10 Hrs.

Classification: Preliminaries, General Approach To Solving Classification Problem, Decision Tree Based Classifier, Rule Based Classifier, Nearest Neighbour Classifier. **Cluster Analysis**– Partitioning Methods: k-Means and k- Mediods– Hierarchical Methods: Agglomerative and Divisive. Density–Based Method: DBSCAN Model Based Clustering Methods Clustering High Dimensional Data Outlier Analysis. **Data mining trends and research frontiers**: Mining complex data types, other methodologies of data mining, Data mining applications, Data Mining and society.

Suggested Learning resources

Text Books:

1. JiaweiHan, Micheline Kamber, JianPei, DataMining Concepts and Techniques, Morgan Kaufmann
2. Pang-ning tan, Michael Steinbach, Vipin Kumar, Introduction to data mining PearsonIN**.**

Course Outcomes:

CO1: Explain the fundamental concepts, scope, and applications of data mining and the importance of data pre-processing.

CO2: Apply suitable data pre-processing techniques such as cleaning, integration, reduction, and transformation on real-world datasets.

CO3: Describe the architecture and design of data warehouses and differentiate OLTP and OLAP operations.

CO4: Implement and analyze data mining techniques such as association rule mining, classification, and clustering on large datasets.

CO5: Summarize emerging trends, research issues, and societal impacts of data mining applications.

Course Outcomes	Programme Outcomes							
	1	2	3	4	5	6	7	8
CO1	1							
CO2		2						
CO3	1		3					
CO4	1							
CO5	1			4				

Course Code: 1PCAB302E	Big Data Analytics	Credits:03
Hrs/Week:L:T:P:S **3:0:0:0**		CIE Marks:50
Total Hours/Week: 03 Hrs		SEE Marks:50

Course Objectives:

1. Understand and define digital data and types, business intelligence, big data and analytics.
2. Appreciate different techniques for big data analytics.
3. Comprehend HDFS architecture and programming environment.
4. Learn NoSQL and able to write statement to process unstructured data.
5. Study programming in Hive and Pig technologies.

UNIT -I	10Hrs.

Types of digital data: Classification of Digital Data, Structured Data, Semi-Structured Data, Unstructured Data. **Introduction to Big Data:** Characteristics, Evolution, Definitions and Challenges of big data, other characteristics of data which are not definitional traits of big data, Why big data? Are we just an information consumer or do we also produce information? Traditional Business Intelligence (BI) versus Big data, A typical Data Warehouse environment, A typical Hadoop environment, What is changing in the realms of big data? **Big data analytics** Where do we begin? What is big data analytics? What big data analytics isn't? Why this sudden hype around big data analytics? Classification of analytics, Top challenges facing big data, why is big data analytics important? Greatest challenges that prevent businesses from capitalizing on big data, what kind of technologies are we looking towards to help meet the challenges posed by big data? Data science, Data Scientist, Terminologies used in big data environment, BASE, Few top Analytics tools.

UNIT -II	10 Hrs.

The big data technology landscape: NoSQL, Where is it used? What is it? Types of NoSQL databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? NoSQL Vendors, SQL Versus NoSQL , NewSQL, Comparison of SQL, NoSQL and NewSQL, **Hadoop:** Features of Hadoop, Key advantages of Hadoop,Versions of Hadoop, Overview of Hadoop Ecosystems, Hadoop Versus, SQL, Integrated Hadoop systems offered by leading market vendors, Cloud based Hadoop solutions. **Hadoop:** Introducing Hadoop, Why not RDBMS, Distributed Computing Challenges, History of Hadoop, Hadoop Overview, Hadoop Components, High Level Architecture of Hadoop, Hadoop Distributed File System(HDFS), HDFS Architecture, Daemons Related to HDFS, Working with HDFS Command, Special Features of Hadoop, Processing Data With Hadoop, Introduction, How Map Reduce Works? Map Reduce Example, Word Count Example using Java. Managing Resources and Applications with YARN, Introduction, Limitation of Hadoop 1.0, Hadoop 2: HDFS, Hadoop 2: YARN, Interacting with Hadoop EcoSystem, Hive,Pig, HBASE, Sqoop, Business Intelligence on Hadoop.

UNIT -III	10Hrs.

NoSQL - MongoDB: What is MongoDB? Why MongoDB? Using JSON, Creating or generating a unique key,

Support for dynamic queries, Storing binary data, Replication, Sharding, Updating information in-place, Terms used in RDBMS and MongoDB, Data types in MongoDB, MongoDB - CRUD (Insert(), Update(), Save(), Remove(), find()), MongoDB- Arrays, Java Scripts, Cursors, Map Reduce Programming, Aggregations. NoSQL - **Cassandra**: What is Cassandra? Why Cassandra? Peer to peer network, Gossip and Failure detection, Anti-Entropy & Read Repair, Writes in Cassandra, Hinted handoffs, Tunable consistency, Cassandra- CQLSH - CRUD, Counter, List, Set, Map, Tracing.

UNIT -IV	**10 Hrs.**

Hadoop Hive: Introduction to Hive - The Problem, Solution - Hive Use Case, Data Growth, Schema Flexibility and Evolution, Extensibility, What is Hive, History of Hive and Recent Releases of Hive, Hive Features, Hive Integration and Work Flow, Hive Data Units, Hive Architecture, Hive Primitive Data Types and Collection Types, Hive File Formats, Hive Query Language - Statements, DDL , DML, Hive Partitions, Bucketing, Views, Sub Query, Joins, Hive User Defined Function, Aggregations in Hive, Group by and Having, Serialization and Deserialization, Hive Analytic Functions. **Hadoop - Pig:** Introducing Pig, History and Anatomy of Pig, Pig on Hadoop, Pig Features, Pig Philosophy, Word count example using Pig, Use Case for Pig, Pig Primitive Data Types, Collection Types and NULL, Pig Latin Overview, Pig Latin Grammar - Comments, Keywords, Identifiers, Case sensitivity in Pig, Common Operators in Pig, Pig Statements, LOAD, STORE, DUMP, Interactive Shell -GRUNT, FILTER, SORT, GROUP BY, ORDER BY, JOIN, LIMIT, Pig Latin Script, Local Mode, Map Reduce Mode, Running Pig Script, Working with Field, Tuple, Bag, User Defined Function, Parameters in Pig.

Reference Books

1. Seema Acharya, Subhashini Chellappan, Big Data and Analytics, Wiley Publications, 2nd Edition, 2019, ISBN:978-81-265-5478-2.
2. Raj Kamal, Preethi Saxena, Big Data Analytics, Introduction to Hadoop, Spark and Machine Learning, McGraw hill Education.
3. Cindi Howson, Successful Business Intelligence, McGraw-Hill Publications, E-ISSN:0-07-149851-6.
4. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch , James Giles, David Corrigan, "Harness the Power of Big data – The big data platform", McGraw Hill, 2012.

Course Outcomes:

CO1: Differentiation of digital data and to define big data and analytics.

CO2: Apply different techniques for big data analytics.

CO3: Comprehend HDFS architecture and programming environment.

CO4: Device NoSQL statement to process unstructured data.

CO5: Understanding programming in Hive and Pig technologies.

Course Outcomes	Programme Outcomes							
	1	2	3	4	5	6	7	8
CO1	1							
CO2		2						
CO3	1		3					
CO4	1							

CO5	1			4				

Course Code: 1PCAB306E	Deep Learning Fundamentals	Credits: 03
L:T:P - 3:0:0		CIEMarks:50
Total Hours/Week: 03		SEEMarks:50

Course Learning objectives:

1. Explain the fundamental concepts of artificial neurons, neural network architectures, and learning paradigms.
2. Apply perceptron learning and backpropagation algorithms to solve classification and pattern recognition problems.
3. Illustrate the strength and weaknesses of many popular deep learning approaches.
4. Analyze different regularization and optimization techniques used in training deep learning models.
5. Develop sequence modeling solutions using recurrent neural networks (RNNs), LSTMs, and encoder-decoder architectures for time-series or sequential data.

UNIT -I	10 Hrs.

Introduction: What is a Neural Network? The Human Brain, Models of a Neuron, Neural Networks Viewed As Directed Graphs, Feedback, Network Architectures Rosenblatt's Perceptron: Introduction, Perceptron, The Perceptron Convergence Theorem, Relation Between the Perceptron and Bayes Classifier for a Gaussian Environment.

UNIT –II	10 Hrs.

Multilayer Perceptrons: Introduction, Some Preliminaries, Batch Learning and On-Line Learning, The Back Propagation Algorithm, XOR Problem, Heuristics for Making the Back- Propagation Algorithm Perform Better, Computer Experiment: Pattern Classification, Back Propagation and Differentiation. Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under Constrained Problem.

UNIT –III	10 Hrs.

Dataset Augmentation, Semi-Supervised Learning. Optimization for Training Deep Models: How Learning Differs from pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rate. Convolution Networks: The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms.

UNIT –IV	10 Hrs.

Random or Unsupervised Features, The Neuroscientific Basic for Convolutional Network, Convolutional Networks and the History of Deep Learning. Sequence Modeling: Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to- Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, The Long Short-Term Memory and Other Gated RNNs.

Text Books

Suggested Learning Resources:

Textbooks

1. Simon Haykin, Neural networks and Learning Machines, Third Edition, Pearson, 2016

2. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016.

Reference Books

Reference book

1. Bengio, Yoshua. "Learning deep architectures for AI." Foundations and trends in Machine Learning, 2009

2. N.D. Lewis, “Deep Learning Made Easy with R: A Gentle Introduction for Data Science”, January 2016

3. Nikhil Buduma, “Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms”, O’Reilly publications

Course Outcomes

CO1: Understanding Deep Learning Fundamentals

CO2: Design and Implementation of Neural Networks

CO3: Optimization and Performance Tuning

CO4: Application of Advanced Deep Learning Architectures

Course Outcomes	Programme Outcomes							
	1	2	3	4	5	6	7	8
CO1	1							
CO2		2		2				
CO3								
CO4			1,2					3
CO5								

1PCAC301E	Block Chain Technologies	Credits: 03
Hrs/Week: L:T:P:S **3:0:0:0**		CIEMarks:50
Total Hours/Week: 40Hrs		SEEMarks:50

UNIT-I	10 Hrs.
Introduction to Blockchain: Introduction to Blockchain, How Blockchain works, Blockchain vs Bitcoin, Practical applications,public and private key basics, pros and cons of Blockchain, Myths about Bitcoin.	
UNIT–II	**10 Hrs.**
Blockchain: Architecture, versions, variants, use cases, Life use cases of blockchain, Blockchain vs sharedDatabase, Introduction to cryptocurrencies, Types, Applications.	
UNIT–III	**10 Hrs.**
Concept of Double Spending: Concept of Double Spending, Hashing, Mining, Proof of work. Introduction to Merkel tree, Privacy,payment verification , Resolving Conflicts , Creation of Blocks.	
UNIT–IV	**10 Hrs.**
Introduction to Bitcoin and Ethereum: Introduction to Bitcoin, key concepts of Bitcoin, Merits and De Merits Fork and Segwits, Sending and Receiving bitcoins, choosing bitcoin wallet, Converting Bitcoins to Fiat Currency. Introduction to Ethereum, Advantages and Disadvantages, Ethereum vs Bitcoin, Introduction to Smartcontracts, usage, application, working principle, Law and Regulations. Case Study.	

Reference Books

1. Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions by ArshdeepBikramaditya Signal, GautamDhameja (PriyansuSekhar Panda., APress.
2. Blockchain Applications: A Hands-On Approach by Bahga, Vijay Madisetti.

Course Outcomes

After completion of the course student will be able to:

1. Demonstrate the basics of Block chain concepts using modern tools/technologies.
2. Analyze the role of block chain applications in different domains including cybersecurity.
3. Evaluate the usage of Block chain implementation/features for the given problem.
4. Exemplify the usage of bitcoins and its impact on the economy.
5. Analyze the application of specific block chain architecture for a given problem.

Course Code: 1PCAC304E	Cloud Essentials	Credits: 03
L:T:P - 3:0:0		CIEMarks:50
Total Hours/Week: 03		SEEMarks:50

Course Learning objectives:

1. Explain the fundamentals of cloud.
2. Analyze Business Benefits and Risks of Cloud Computing.
3. Evaluate Emerging Trends in Cloud Computing.

UNIT-I	10 Hrs.

Introduction to Cloud: Defining a cloud, Characteristics of Cloud Computing, Cloud computing reference model, Architectures for parallel and distributed computing, Elements of parallel computing and Elements of distributed computing. Cloud Service Models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Function as a Service (FaaS), Blockchain-as-a-Service (BaaS) and use cases, Cloud Deployment Models: Public Cloud, Private Cloud, Hybrid and Multi-Cloud, Community Cloud..

UNIT–II	10 Hrs.

Core Components of Cloud Architecture: Compute Services, Storage Services, Networking Services, Virtualization, Types of Virtualization, Containers vs. Virtual Machines (VMs), Load Balancing in Cloud, Auto-Scaling & Fault Tolerance, Content Delivery Networks(CDN), Bare Metal Cloud ,Cloud Orchestration and Automation..

UNIT–III	10 Hrs.

Cloud Automation, DevOps, and Future Innovations: Cloud Automation and Infrastructure as Code (IaC), DevOps and Continuous Integration/Continuous Deployment (CI/CD), Multi-Cloud and Hybrid Cloud Strategies, Sustainability and Green Cloud Computing, Cloud Innovations- AI-powered cloud automation-5G and its impact on cloud computing..

UNIT–IV	10 Hrs.

Cloud Security & Risk Management: Cloud Adoption, Advantages and Challenges of Cloud Adoption, Security Risks in Cloud- Data Breaches, Identity Theft, Network Security in Cloud. Security Solutions in Cloud: Identity and Access Management (IAM), Data Encryption Techniques, Firewalls & Intrusion Detection Systems: Compliance & Regulatory Frameworks, Disaster Recovery and Business Continuity Planning in Cloud. **Emerging Trends and case study**: AI, Edge Computing, Quantum Cloud, Event-driven architecture in cloud, Cloud-based AI services -AWS SageMaker, Google Vertex AI, Azure ML. Case Study: Netflix's Cloud Migration, Zoom's Cloud Scalability

Text Books

1. CompTIA Cloud Essentials+ Study Guide: Exam CLO-002 [2 ed.] .

2. Rajkumar Buyya, Christian Vecchiola, and Thamrai Selvi Mastering Cloud Computing McGraw Hill Education.

3. Handbook of Cloud Computing, Borko Furht- Armando Escalante.

4. Cloud Essentials: CompTIA Authorized Courseware for Exam CLO-001.

5. RjkumarBuyya, Christian Vecchiola, and ThamaraiSelci, Mastering Cloud Computing, Tata McGraw Hill, New Delhi, India, 2013.

Reference Books

1. Cloud Computing for Dummies by Judith Hurwitz, R.Bloor, M. Kanfman, F.Halper (Wiley India Edition)
2. Toby Velte, Anthony Velte, Cloud Computing: A Practical Approach, McGraw-Hill Osborne Media

Course Outcomes

CO1: Demonstrate the fundamental concepts of cloud computing.

CO2: Understand Cloud Security Challenges

CO3: Understand and Explain Cloud Compute Services and Analyze Cloud Networking Services

CO4: Compare different deployment and service models of cloud to develop different variety of applications with securities

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1	2							
CO2	2	2			2			
CO3	2	2			3			
CO4	2	3		3				3
CO5								

Course Code:1PCAC305E	Cyber Security	Credits: 03
L:T:P - 3:0:0		CIEMarks:50
Total Hours/Week: 03		SEEMarks:50

Course objectives:

1. To learn cybercrime and cyber law
2. To understand the cyber-attacks and tools for mitigating them.
3. To understand information gathering.
4. To learn how to detect a cyber-attack.
5. To learn how to prevent a cyber-attack.

UNIT -I — 10 Hrs.

INTRODUCTION Cyber Security – History of Internet – Impact of Internet – CIA Triad; Reason for Cyber Crime – Need for Cyber Security – History of Cyber Crime; Cyber criminals – Classification of Cyber crimes– A Global Perspective on Cyber Crimes; Cyber Laws – The Indian IT Act –Cyber crime and Punishment.

UNIT –II — 10 Hrs.

ATTACKS AND COUNTER MEASURE SOS WAP : Malicious Attack Threats and Vulnerabilities: Scope of Cyber-Attacks – Security Breach – Types of Malicious Attacks – Malicious Software Common Attack Vectors – Social engineering Attack – Wireless Network Attack – Web Application Attack – Attack Tools – Counter measures.

UNIT –III — 10 Hrs.

RECONNAISSANCE Harvester – Who is Netcraft–Host–Extracting Information from DNS – Extracting Information from E-mail Servers – Social Engineering Reconnaissance; Scanning – Port Scanning – Network Scanning and Vulnerability Scanning – Scanning Methodology – Ping Sweer Techniques – Nmap Command Switches.

UNIT –IV — 10 Hrs.

INTRUSION DETECTION :Host -Based Intrusion Detection – Network -Based Intrusion Detection– Distributed or Hybrid Intrusion Detection – Intrusion Detection Exchange Format – Honeypots – Example System Snort.

INTRUSION PREVENTION: Firewalls and Intrusion Prevention Systems: Need for Firewalls – Firewall Characteristics and Access Policy – Types of Firewalls – Firewall Basing – Firewall Location and Configurations –Intrusion Prevention Systems – Example Unified Threat Management Products.

Text Books

1. Patrick Engebret son,—The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made easy‖, Elsevier, 2011. (Unit-3)
2. William Stallings, LawrieBrown,—Computer Security Principles and Practice‖,ThirdEdition, Pearson Education, 2015. (Unit-4 & 5)
3. Anand Shinde, —Introduction to Cyber Security Guide to the World of Cyber Security‖, Notion Press, 2021. (Unit-1 &2)

Reference Books

1.David Kim, Michael G. Solomon, —Fundamentals of Information Systems Security‖, Jones & Bartlett Learning Publishers, 2013.
2. Nina Godbole, Sunit Belapure,— Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives‖, Wiley Publishers, 2011.

Course Outcomes

CO1: Explain the basics of cyber security, cybercrime and cyber law

CO2: Classify various types of attacks and learn the tools to launch the attacks

CO3: Apply various tools to perform information gathering

CO4: Apply intrusion techniques to detect intrusion

CO5: Apply intrusion prevention techniques to prevent intrusion

Course Outcomes	Programme Outcomes							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
CO1		1						
CO2		2		3				
CO3	2	2			3			
CO4	2	2			3	3		
CO5		1		2			3	