St Joseph's University

Bengaluru

Re-accredited by NAAC "A++" Grade

Course Structure and Syllabus

of

Master of Computer Application

(M.C.A.)

CHOICE BASED CREDIT SYSTEM

(Student admitted from 2025 – 2026 onwards)

SCHEME OF EXAMINATION

FIRST SEMESTER

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CA 7125	Database Design and Implementation	4	4	50	50	100
2	CA 7225	Data Structures and Analysis of Algorithms	4	4	50	50	100
3	CA 7325	Machine Learning with Python	4	4	50	50	100
4	CA 7425	Mathematical Foundations for Computer Applications	3	3	50	50	100
5	CA 7P125	Data Structures and Algorithms Lab	2	4	25	25	50
6	CA 7P225	DBMS Lab	2	4	25	25	50
7	CA 7P325	Machine Learning with Python Lab	2	4	25	25	50
8	CA 7525	Finance, Accounting, and Company Laws for IT Professionals	3	3	50	50	100
9	CA 7625	Entrepreneurship and Innovation	2	2	25	25	50
10		**IGNITORS I**	1	--	--	--	--
		Total	**27**	**32**	**350**	**350**	**700**

SECOND SEMESTER

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CA 8125	Information and Network Security	4	4	50	50	100
2	CA 8225	Advanced Programming with Java	4	4	50	50	100
3	CADE 8325 & CADE 8425	**ELECTIVE I (Theory)**	4	4	50	50	100

4	CADE 8525 & CADE 8625	**ELECTIVE II (Theory)**	4	4	50	50	100
5	CA 8P125	Advanced Java Lab	2	4	25	25	50
6	CADE 8P225 & CADE 8P325	**ELECTIVE I (Practical)**	2	4	25	25	50
7	CADE 8P425 & CADE 8P525	**ELECTIVE II (Practical)**	2	4	25	25	50
8	CA 8725	Internet of Things	3	3	50	50	100
9		**OUTREACH**	2	--	--	--	--
		Total	**27**	**31**	**325**	**325**	**650**

THIRD SEMESTER

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CA 9125	Agile Software Engineering	4	4	50	50	100
2	CA 9225	Cloud Computing	4	4	50	50	100
3	CA 9325	Full Stack Development	4	4	50	50	100
4	CA 9425	Deep Learning	4	4	50	50	100
5	CA 9P125	Deep Learning Lab	2	4	25	25	50
6	CA 9P225	Cloud Computing Lab	2	4	25	25	50
7	CA 9P325	Full Stack Development Lab	2	4	25	25	50
8	CA 9525	Research Methodology	3	3	50	50	100
9		**IGNITORS II**	1	--	--	--	--
		Total	**26**	**31**	**325**	**325**	**650**

FOURTH SEMESTER

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CA 10PR1	Industry Internship / Project Work	16	3	200	200	400
2	CA 10CC1	Online / Certification Courses	3	3	50	50	100
		Total	**19**	**6**	**250**	**250**	**500**

ELECTIVE I (Theory)

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CADE 8325	Natural Language Processing	4	4	50	50	100
2	CADE 8425	Image Processing and Pattern Recognition	4	4	50	50	100

ELECTIVE I (Practical)

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CADE 8P325	Natural Language Processing Lab	2	2	25	25	50
2	CADE 8P425	Image Processing Lab	2	2	25	25	50

ELECTIVE II (Theory)

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CADE 8525	Big Data Analytics	4	4	50	50	100
2	CADE 8625	Data Science	4	4	50	50	100

ELECTIVE II (Practical)

Sl No	Subject Code	Subject Title	Credits	Number of Hours per week	Continuous Internal Assessment (CIA) Marks	End Semester Marks	Total Marks
1	CADE 8P525	Big Data Analytics Lab	2	2	25	25	50
2	CADE 8P625	Data Science Lab	2	2	25	25	50

FIRST SEMESTER

CA 7125: DATABASE DESIGN AND IMPLEMENTATION

Total No. of Lectures: 60	**Total Marks : 100**	**[L – T – P – S]**
No. of Lectures / Week: 4	**Credits : 4**	**[3 – 1 – 0 – 2]**

Course objectives:

Upon successful completion of this course, students should be able to:

- Understand the limitations of traditional file management systems.
- Understand the need for an efficient management system to administer the data repository of any organization.
- Importance of data consistency and also how data integrity ignorance affects any business organization.
- Providing data security through different Views.
- Identifying the power of Structured Query language, which generates flexible and customized reports.
- Providing complex integrity constraints through the use of Triggers.
- Know the power of procedural SQL, writing Stored procedures, functions, and packages.

Course Outcomes:

After successful completion of this course, the students should:

- Have a good understanding of data and database systems and their design issues.
- Describe the fundamental elements of relational database management systems.
- Understand the design of relational databases through the use of Entity-Relationship Diagrams and Normalization procedures.
- Develop basic skills in the use of SQL in defining and creating a database and inserting and modifying entries in a table.
- Have an Effective way of manipulating the database to produce useful decision-making information for management & analytics.
- Handling the distributed database and understanding the issues related to data distribution
- Understand the concept of Big data domain and Schema-less storage.

UNIT I

Introduction-Database System Applications: Purpose of Database Systems, View of Data – Data, Data Abstraction, Three Schema Architecture, Data Independence, Database Users and Administrators, Data Models.

Introduction to Data model: ER model: Entities, Attributes and Entity sets, relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises.

Relational Model: Foundations of Relational model, Basics of Relational Algebra and Calculus.

(08 Hours)

UNIT II

Relational algebra: Introduction, Basic Operations: Union, Intersection, Cartesian Product, Difference, **Relational Algebraic Operations:** Projection, Join, Selection and Division, **Relational model constraints:** Domain constraints, Key constraints, Entity integrity, referential integrity

Normalization: Concept of normalization, Need for Normalization, Anomalies in DB, Normalization Process, Functional Dependency, Full Functional Dependency, Partial Dependency, trivial functional dependency, 2NF, 3NF, BCNF, 4NF and 5NF, Join and Multivalued Dependency. Denormalization, Database design with Normal forms. **(13 Hours)**

UNIT III

SQL: Components of SQL: Data Definition Language, Data Manipulation Language, Data Control Language, Data Query Language, Transaction Control Language

Creating Tables, Altering Tables and Schema, Manipulation of Data, Querying Single Table and Multiple Tables, Adding /Altering Constraints and Business Rules, Sorting, Join Operations, Data Dictionary.

Views: Concept, purpose of views, types of views, Advantages and disadvantages of views, implementing views in SQL, Altering Views

Nested queries: Features of subquery, Rules, types of sub-query

Set operators in SQL: Using Union, Union All, Intersect, and Minus

Functions: Scalar and aggregate functions: types of scalar functions, Aggregate functions, Conversion functions **(14 Hours)**

UNIT IV

File Organization: Heap Files, Sequential Access Methods, Indexed Sequential Access Method, Indexes: Purpose Index, Types of Indexes, Advantages and Disadvantages of creating Indexes, Multi-level indexes, Creating Indexes in SQL, purpose of Index, types of indexes, Implementing Indexes in SQL, Sequences

Transactions: Concept of Transaction, States of Transaction, ACID properties, **Concurrent Transaction:** Lost update, Dirty Read, Unrepeatable Read, Inconsistent summary, Concurrency control Mechanism: Database Lock, 2 PL: Basic 2 PL, Strict 2 PL, Rigorous 2 PL, Timestamp: Deadlock, deadlock prevention algorithms: Wait-Die and Wound-Wait

Failures in DB/Transaction: Reasons for Database failures, Catastrophic failure, Transaction Log, Backup, transaction Failure, Deferred update, Immediate update, Log-based recovery

PL/SQL: A Programming Language: History – Fundamentals – Block Structure – Comments – Data Types, Assignment operation – Bind variables, Conditional statement, Branching and Looping, **Transaction Control statements**, DCL: Commit and Rollback **(13 Hours)**

UNIT V

PL/SQL Cursors and Exceptions: Cursors – Implicit & Explicit Cursors and Attributes – Cursor FOR loops, Cursor with Parameters – Cursor Variables – Exceptions – Types of Exceptions. Stored Procedures and Triggers

Distributed Database Systems: Components, Advantages over centralized database system, Transparencies in DDB, Issues in DDB: Fragmentation, Replication and Allocation, DDB architecture: SPSD, MPSD, MPMD, Distributed query processing, 3-phase commit protocol

Big data: Concept of Big Data, 4 V's of Big Data, ACID vs BASE, NoSQL, Types of NoSQL Data.

Vector Databases: Introduction to Vector Databases – Applications in AI. **(12 Hours)**

Text Books

[1] Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, Seventh Edition, McGraw-Hill Education, 2020.

[2] R. Elmasri and S.B. Navathe: "Fundamentals of Database System", Pearson, 7th Edition, 2018.

[3] Nilesh Shah, "Database Systems using ORACLE: A simplified guide to SQL, and PL/SQL", 2nd Edition, Prentice Hall of India Ltd.

[4] Akmal Chaudhri, Arnaud Comet, Eric Hanson, Madhukar Kumar, Vector Databases and AI Applications for Dummies, SingleStore Special Edition, 2025

Reference Books:

[1] Raghurama Krishnan, Johannes Gehrke,Data base Management Systems, 3rd Edition, 2014, McGrawHill Education,

[2] Bipin C. Desai, "Introduction to Database Systems", Galgotia Publications.

[3] Date, C. J. , "An introduction to database systems", 3rd Edition, Narosa publishing house.

[4] Hansen & Hansen, "Database Management and Design", Prentice Hall of India ltd.

[5] Ullman, J. D., "Principals of Database systems", Galgotia publications

[6] Narang, "Database Management System", Prentice Hall of India Ltd

[7] Ivan Bayross "SQL, PL/SQL : The programming language of Oracle", BPB Publications

CA 7225: DATA STRUCTURES AND ANALYSIS OF ALGORITHMS

Total No. of Lectures: 60 **Total Marks: 100** **[L – T – P – S]**
No. of Lectures/Week: 4 **Credits: 4** **[3 – 1 – 0 – 2]**

Course Objectives:

- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyse the Performance of algorithms.
- To understand the behaviour of data structures such as stacks, queues, trees, hash tables, search trees, Graphs and their representations.
- To choose an appropriate data structure for a specified application.
- To understand and analyse various searching and sorting algorithms.

Course Outcomes:

- Ability to choose appropriate data structures to represent data items in real world problems.
- Ability to analyse the time and space complexities of algorithms.
- Ability to design programs using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, and B-trees.
- Able to analyse and implement various kinds of searching and sorting techniques.

UNIT I

Introduction: Introduction to structures, functions, Pointers, Memory Allocation -. Introduction to one dimensional array, two dimensional array, character array – string operations, sparse matrix and its operations . What is data structure, different types of data structures, Application of data structures **(07 Hours)**

UNIT II

Stack: stack as an ADT, Representing Stack using C, Stack Applications – infix to postfix, postfix evaluation, Decimal to binary conversion, Recursion

Queue: Queue as an ADT, Queue implementation using C, Queue Applications, Queue Types – Circular Queue, Double Ended Queue, Priority Queue – Implementation using array.

Lists: Introduction to lists, Types of linked list, Node Structure, list operations – insert and delete, circular linked list – insert and delete operations, Double ended and double ended circular queue – insert and delete operations. Representation of Exponential equation and addition of exponential equations. Ordered singly linked list. **(15 Hours)**

UNIT III

Trees: Introduction to trees, Binary trees – Operations on binary trees, Applications of Binary trees – duplicate value deletion, expression representation and traversal. Binary tree

representation – Node Representation of binary trees, implicit array representation of binary trees, binary tree traversals, Threaded binary trees. Binary Search trees. Self-balancing trees -AVL trees., B trees, splay trees, Huffman Coding, Quad tree, 2-3 tree, Red Black trees. **(15 Hours)**

UNIT IV

Searching: Linear Search, Binary Search, Hashing – insertion and searching using hashing.

Sorting: Bubble sort, insertion sort, selection sort, radix sort, merge sort, quick sort, heap sort, address calculation sort.

Graphs: Definitions, Graph Representations- Adjacency matrix, Adjacency lists, warshall's algorithm, shortest path algorithm, **Graph traversal methods:** Depth First Search and Breadth First Search, Graph spanning tree – prims algorithm and kruskal's algorithm. **(12 Hours)**

UNIT V

Analysis of Algorithms: Definition of an Algorithm, Characteristics of a Good Algorithm, Asymptotic Notations, Time and Space Complexity, Growth of Functions & Complexity Classes, **Complexity Analysis Examples:** Calculation of the complexity of recursive algorithms, Brute Force, Divide and Conquer, Greedy, NP-Completeness & Approximation Algorithms. **(11 Hours)**

TEXT BOOKS:

[1] Data structures, using C and C++, 2nd Edition, Yedidyah Langsam, Aaron M. Tenenbaum. Pearson

[2] Design and Analysis of Algorithms By A.A.Puntambekar Technical Publications, 2010.

REFERENCE BOOKS:

[1] DATA STRUCTURES THROUGH C IN DEPTH by S. K. srivastava and Deepali Srivastav BPB Publications.

[2] Data Structures – R Venkateshan, S. Lovelyn rose – Second Edition – Wiley

[3] DATA STRUCTURES AND ALGORITHMS MADE EASY by Narasimha Karumanchi - Careermonk Publications; 5th ed. edition (2016)

[4] Data structures and Algorithms in C++, M.T. Goodrich, R. Tamassia and D. Mount, Wiley India.

CA 7325: MACHINE LEARNING WITH PYTHON

Total No. of Lectures: 60	**Total Marks: 100**	**[L – T – P – S]**
No. of Lectures / Week: 4	**Credits : 4**	**[3 – 1 – 0 – 2]**

Course objectives:

The objectives of this course are to:

- Introduce the fundamental concepts of Python programming, including syntax, control flow, and error handling.

- Enable students to design reusable and maintainable code using functions and object-oriented programming principles.
- Familiarize students with the basics of machine learning, emphasizing data preparation and Python libraries such as NumPy, Pandas, and Matplotlib.
- Equip students with the ability to implement and evaluate supervised learning models for predictive tasks.
- Provide hands-on experience with unsupervised learning methods, focusing on clustering, dimensionality reduction, and applications like anomaly detection and customer segmentation.

Course Outcomes:

Upon completing this course, students will be able to:

- Understand the fundamentals of Python programming and apply basic syntax, data types, control flow, and operators for writing efficient code.
- Develop modular programs using functions and implement object-oriented programming concepts such as classes, inheritance, and polymorphism.
- Demonstrate knowledge of machine learning concepts and apply Python libraries for data preprocessing, feature engineering, and visualization.
- Implement supervised learning algorithms such as regression and classification while evaluating models using appropriate metrics.
- Explore and apply unsupervised learning techniques such as clustering and dimensionality reduction to solve real-world problems.

UNIT I

Python for Machine Learning: Getting Started with Python: Basic Syntax, Interactive Shell, Editing, Saving, and Running Scripts. **Programming Concepts: Data Types:** List, String, tuple, dictionary, set, Slicing and Indexing, **Control Flow & Data Processing:** Conditional Statements, Looping Constructs, Break, Continue, and Pass Statements, Functions and Modules, Understanding packages, Importing packages. File Handling (Reading/Writing Files, CSV, JSON) **(12 Hours)**

UNIT II

Data Handling and Visualization: NumPy for Scientific Computing: NumPy Arrays and Basic Operations, Indexing, Slicing, and Reshaping Arrays, Mathematical Functions and Linear Algebra in NumPy. **Pandas for Data Manipulation:** Creating and Managing DataFrames, Filtering, Sorting, and Aggregating Data, Handling Missing Values and Duplicates, Merging and Concatenating DataFrames. **Data Visualization using Matplotlib & Seaborn:** Creating Line, Bar, and Scatter Plots, Heatmaps, Histograms, Box Plots for Data Analysis, Interactive Visualizations with Seaborn. **(12 Hours)**

UNIT III

Data Preprocessing and Feature Engineering: Data Cleaning & Handling Missing Data: Identifying and Handling Missing Values, Dropping or Imputing Data Using Statistical Methods. **Feature Engineering:** Encoding Categorical Variables (One-Hot, Label Encoding), Scaling and Normalization (Min-Max Scaling, Standardization), Feature Selection Techniques. **Dimensionality Reduction Techniques:** Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA). **(14 Hours)**

UNIT IV

Supervised Learning & Model Evaluation: Regression Models: Simple & Multiple Linear Regression, **Performance Metrics:** Mean Squared Error (MSE), RMSE, R^2. **Classification Models:** Logistic Regression, k-Nearest Neighbors (k-NN), Naïve Bayes Classifier. **Model Evaluation Techniques:** Train-Test Split & Cross-Validation, Confusion Matrix, Precision, Recall, F1-Score, ROC Curve and AUC Score. **(13 Hours)**

UNIT V

Unsupervised Learning & Applications: Clustering Techniques: K-Means Clustering, DBSCAN (Density-Based Spatial Clustering), Hierarchical Clustering. Real-World Applications of **Unsupervised Learning:** Customer Segmentation in Marketing, Anomaly Detection for Fraud Detection, Recommender Systems. **(09 Hours)**

Text Books

[1] Python Programming: A Modern Approach, Vamsi Kurama, Publisher: Pearson Education

[2] Machine Learning with Python Cookbook , Chris Albon, O'Reilly Media

[3] Introduction to Machine Learning with Python, Andreas C. Müller and Sarah Guido, O'Reilly Media

[4] Data Science from Scratch: First Principles with Python, Joel Grus, O'Reilly Media

[5] Python Machine Learning, Sebastian Raschka and Vahid Mirjalili, Packt Publishing

Reference Books:

[1] Think Python: How to Think Like a Computer Scientist, Allen B. Downey, O'Reilly Media

[2] Python for Data Analysis, Wes McKinney, O'Reilly Media

[3] Programming Python, Mark Lutz, O'Reilly Media

[4] Pattern Recognition and Machine Learning, Christopher M. Bishop, Springer

[5] An Introduction to Statistical Learning, Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, Publisher: Springer

[6] Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig, Pearson Education

CA 7425: MATHEMATICAL FOUNDATIONS FOR COMPUTER APPLICATIONS

Total no of lectures: 42 **Total marks: 100** **[L – T – P – S]**

No of lectures / week: 3 **Credits: 3** **[3 – 1 – 1 – 1]**

Course objectives:

- To equip students with the essential mathematical foundations in set theory, logic, probability, statistics, and linear algebra, which are fundamental for machine learning.
- To provide students with the tools necessary to analyze and manipulate data using probability distributions and statistical methods.
- To enhance students' ability to apply probability theory and statistical inference in real-world machine learning problems.
- To introduce advanced concepts like stochastic processes, Bayes' theorem, and Monte Carlo methods, which are critical for machine learning models.
- To develop the skills needed for model evaluation, optimization, and interpretation using statistical methods like regression analysis and PCA.

Course Outcomes:

After completing this course, students will be able to:

- Understand and apply core mathematical concepts in machine learning tasks, including set theory, logic, and linear algebra.
- Able to calculate and interpret probabilities, random variables, and key statistical measures such as mean, variance, and correlation.
- Gain proficiency in statistical inference, including hypothesis testing, confidence intervals, and regression models.
- Apply advanced probability techniques, including Bayes' theorem, stochastic processes, and Monte Carlo methods, to solve machine learning problems.
- Use statistical methods like PCA, ANOVA, and cross-validation to evaluate, optimize, and interpret machine learning models.
- Develop the ability to apply mathematical tools to analyze data and enhance the performance of machine learning algorithms.

UNIT I

Set Theory - Basics of sets, subsets, operations on sets, and Venn diagrams, Cartesian products and power sets. Propositional Logic - Propositions and logical connectives, Truth tables and logical equivalence, Tautologies, contradictions, implications. Normal Forms - Conjunctive Normal Form (CNF), Disjunctive Normal Form (DNF). Predicates and quantifiers, Introduction to rules of inference. Introduction to Proof Techniques- Direct, indirect, contradiction, and mathematical induction **(08 Hours)**

UNIT II

Relations - Definition and types of relations- reflexive, symmetric, transitive, Equivalence relations. Functions- Types of functions: one-to-one, onto, bijective, Composition of functions, and inverse functions. Linear Algebra - Matrix operations, Transpose, determinants, and inverse of matrices, solving systems of linear equations using Gaussian elimination, Eigenvalues, and eigenvectors and their role in machine learning (**08 Hours)**

UNIT III

Probability Theory - Axiomatic definition of probability, sample spaces, and events, Joint and conditional probabilities, independence, Total probability and Bayes' rule. Random Variables- Discrete and continuous random variables, Probability mass function (pmf) and probability density function (pdf), Cumulative distribution function (cdf), Expectation, variance, and moments of random variables **(08 Hours)**

UNIT IV

Probability Distributions, Applications, Uniform, Gaussian, binomial, Poisson distributions, Law of Large Numbers and the Central Limit Theorem (CLT), Markov Chains and their applications in machine learning, Conditional probability and Bayes' Theorem in machine learning contexts. Stochastic Processes,

Basic Statistical Concepts - Descriptive Statistics, Measures of central tendency (mean, median, mode) and dispersion (variance, standard deviation), Scatter plots and histograms for data visualization. Understanding covariance, correlation, and their interpretation in datasets, Introduction to simple linear regression **(09 Hours)**

UNIT V

Statistical Inference - Point and interval estimation, Hypothesis testing (Z-test, T-test, and Chi-square test), Confidence intervals and significance levels. Regression and Model Evaluation metrics**,** R-squared, Adjusted R-squared, MSE, MAE.

Advanced Statistical Methods for ML **-** Analysis of Variance (ANOVA), Principal Component Analysis (PCA) for dimensionality reduction, Logistic regression and its connection to machine learning algorithms **(09 Hours)**

Text Books

1. "Discrete Mathematics and Its Applications" by Kenneth H. Rosen
2. "Linear Algebra and Its Applications" by David C. Lay
3. "Probability and Statistics for Engineering and the Sciences" by Jay L. Devore
4. "Pattern Recognition and Machine Learning" by Christopher M. Bishop

Reference Books:

1. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Academic Press, (2009).
2. D. C. Montgomery and G.C. Runger, "Applied Statistics and Probability for Engineers", 5th edition, John Wiley & Sons, (2009).
3. Robert H. Shumway and David S. Stoffer, "Time Series Analysis and Its Applications with R Examples", Third edition, Springer Texts in Statistics, (2006).

CA 7P125: DATA STRUCTURES AND ALGORITHMS LAB

1. Write and execute MENU DRIVEN program to show the working of a STACK.
2. Write a program in C using a STACK to check whether a given string is a palindrome.
3. Write a program in C using a stack to find the first N FIBONACCI numbers.
4. Write a program in C using a STACK to check whether a given set of brackets are in order.
5. Write a program in C using a STACK to **convert INFIX expression to POSTFIX EXPRESSION.**
6. Write a program in C using a STACK to convert to evaluate a POSTFIX EXPRESSION.
7. Using the above two programs write a program to evaluate an INFIX mathematical expression.
8. Write a menu driven program to create a **double stack** using one array. Show how you can input and pop data items from each of these stacks.
9. Write a **COMMAND LINE** program in C read a text file (.C file) and check for balancing of various parentheses ((), [], { }).
10. Write a **command line program** in C to convert a decimal number into its **BINARY EQUIVALENT** using a stack.
11. Write a program in C with function sub programs to SORT some numbers in an array using **SELECTION SORT**./Also write **a function** sub program to sort numbers using selection sort.
12. Write a program in C with **function sub programs** to search for a given number in an array using **LINEAR SEARCH** method. Let the function return 1 it the number is found or else return -1
    PROTOTYPE: **int fnSearchNum(int N, int A[20], int SearchNum)**
13. Write a program in C with **function sub programs** to search for a given number in an array using **BINARY SEARCH** method. Let the function return 1 it the number is found or else return -1
    PROTOTYPE: **int fnBinarySearch(int N, int A[20], int SearchNum)**
14. Write a function sub program in C to sort some numbers using **INSERTION SORT** technique. Use it in the main program to sort a list f numbers using insertion sort.
15. Write **function sub programs** to a) return the length of a string b) to make a **duplicate copy** of a string c) to **join two** strings d) to compare two string d) To **check whether a given character** is present in a string e) to convert a string to **UPPER case/LOWER case** f) to **reverse a string**. Use them in the main program to show the working of each of these functions.
16. Write a function sub program to sort a list of strings using Bubble sort. (use strcpy() and strcmp()) Use it in the main program to sort a list of names.
17 Write a menu driven program to show the working of an **ORDERED Linked list** with options: 1) Insert a number 2) To delete the first number 3) To delete the last number 4) to delete any number 5) To display the Content of the list 6) To return the total number of nodes in the list 7) To find the average of all the numbers in the list.

18 Write a menu driven program to show the working of a **Queue** with options to 1) Enqueue 2) Dequeue 3) Display the content of the queue.
19 Write a menu driven program to show the working of a **CIRCULAR QUEUE**.
20 Write a program in C to show the working of a **DOUBLY LINKED LIST.**
21 Write a program in C to find the sum of **two polynomials** .
22 Write a program in C to sort a list of numbers using **QUICK SORT** technique.
23 Write a program in C to sort a list of numbers using **MERGE SORT** technique.
24 Write a program in C to do the **Breadth first search**(BFS) of a given graph.
25 Write a program in C to do the **Depth first search(DFS)** of a given graph.
26 Write a program in C to find the **shortest path** using Warshall-Floyd Algorithm(Dynamic Programming)
27 Write a program in C to implement PRIM'S ALGORITHM to find the Minimum Spanning tree for a given graph.
28 Write a program in C to sort a list of numbers using HEAP SORT.
29 Write a program in C sort a list of number using RADIX **SORT**.
30 Write a program in to sort a list of numbers using **ADDRESS CALCULATION SORT**

CA 7P225: DBMS LAB

1. Implementation of DDL and DML Commands of SQL with suitable examples.
2. Implementation of DQL and DCL Commands of SQL with suitable examples.
3. Implement different types of built-in functions in SQL with suitable examples.
4. Implementation of different types of operators in SQL.
5. Implementation of different types of Joins in SQL.
6. Implementation of materialized and non-materialized views in SQL.
7. Implementation of different types of views and nested subqueries in SQL.
8. Implementation of PL/SQL Blocks.
9. Implementation of Exception Handling in PL/SQL.
10. Implementation of PL/SQL Procedures and Functions.
11. Implementation of PL/SQL Packages
12. Implementation of PL/SQL Triggers

CA 7P325: MACHINE LEARNING WITH PYTHON LAB

1. Python Basics
   (i) Write a Python program to demonstrate basic syntax, variables, data types, and operators.
   (ii) Implement a program to use control flow statements (if-else, loops) and functions.
   (iii) Perform file handling operations: read, write, and modify CSV and JSON files.
2. NumPy and Pandas for Data Handling
   (i) Write a Python program using NumPy to perform array creation, reshaping, indexing, and mathematical operations.
   (ii) Implement data handling using Pandas: create DataFrames, filter, sort, and handle missing values.
   (iii) Merge, concatenate, and group DataFrames using Pandas.
3. Data Visualization with Matplotlib and Seaborn
   (i) Create a line plot, bar chart, and scatter plot using Matplotlib.
   (ii) Implement data visualization with Seaborn: histogram, box plot, heatmap.
   (iii) Generate an interactive visualization using Seaborn.
4. Data Preprocessing and Feature Engineering
   (i) Implement techniques for handling missing values (dropping/imputation).
   (ii) Perform categorical variable encoding using One-Hot and Label Encoding.
   (iii) Apply scaling techniques (Min-Max Scaling, Standardization).

(iv) Implement Principal Component Analysis (PCA) for dimensionality reduction.

5. Supervised Learning - Regression
   (i) Implement Simple and Multiple Linear Regression using Scikit-Learn.
   (ii) Evaluate regression models using MSE, RMSE, and R^2 score.
   (iii) Visualize regression model performance with Matplotlib.
6. Supervised Learning - Classification
   (i) Implement Logistic Regression and k-Nearest Neighbors (k-NN) for classification tasks.
   (ii) Use Naïve Bayes Classifier for text classification (e.g., spam detection).
   (iii) Evaluate classification models using Confusion Matrix, Precision, Recall, and F1-score.
7. Model Evaluation & Cross-Validation
   (i) Implement train-test split and k-fold cross-validation.
   (ii) Generate an ROC Curve and compute AUC score for classification models.
8. Unsupervised Learning - Clustering
   (i) Implement K-Means Clustering and visualize clusters.
   (ii) Perform DBSCAN clustering and compare results with K-Means.
   (iii) Implement Hierarchical Clustering and visualize the dendrogram.
9. Real-World Applications
   (i) Implement customer segmentation using clustering techniques.
   (ii) Perform anomaly detection for fraud detection.
   (iii) Build a simple recommendation system using collaborative filtering.
10. Mini-Project
   (i) Apply machine learning techniques to a real-world dataset (e.g., Titanic dataset, Boston Housing dataset).
   (ii) Perform end-to-end model development: data preprocessing, feature engineering, model training, evaluation, and visualization.

CA 7525: FINANCE, ACCOUNTING, AND COMPANY LAWS FOR IT PROFESSIONALS

Total no of lectures: 42	**Total marks: 100**	**[L – T – P –S]**
No of lectures / week: 3	**Credits: 3**	**[3 – 1 – 0 – 1]**

Course Objectives:

- Familiarize students with essential financial concepts to enable effective decision-making in IT projects.
- Introduce the accounting fundamentals, enabling students to analyze and interpret financial statements.
- Provide an understanding of key aspects of company laws relevant to IT professionals, including compliance and legal frameworks.
- Equip students with practical tools and techniques to manage finances, budgets, and contracts in real-world IT scenarios.

UNIT I

Basics of Finance: Introduction to Finance: Role and Importance of Finance in IT, Goals of **Financial Management:** Profit Maximization vs. Wealth Maximization.

Time Value of Money: Concepts: Present Value (PV) and Future Value (FV). **Applications:** Net Present Value (NPV) and Internal Rate of Return (IRR) in IT projects.

Sources of Finance for IT Startups: Equity, Debt, Angel Investors, Venture Capital, and Crowdfunding.

Key Financial Ratios: Liquidity Ratios: Current Ratio, Quick Ratio. Profitability Ratios: Gross Margin, Return on Investment (ROI). **Solvency Ratios:** Debt-to-Equity Ratio.

Working Capital Management: Managing Cash Flows and Short-term Funds for IT projects. **Practical:** Develop a simple cash flow analysis for an IT startup. Calculate NPV and IRR for a hypothetical IT project using spreadsheet software. **(12 Hours)**

UNIT II

Fundamentals of Accounting: Accounting Basics: Key Concepts: Double-Entry System, Accrual vs. Cash Accounting. Fundamental Accounting Equation: Assets = Liabilities + Equity.

Preparation of Financial Statements: Structure and Interpretation of Balance Sheet, Income Statement, and Cash Flow Statement. Relationship between the three statements.

Accounting Cycle: Journal Entries, Ledger Posting, Trial Balance, Adjustments, and Final Accounts.

Cost Accounting: Cost Types: Fixed Costs, Variable Costs, and Semi-Variable Costs. Marginal Costing and Break-even Analysis for IT Projects.

Budgeting for IT Projects: Creating and Managing Budgets. Variance Analysis: Identifying deviations from planned budgets.

Practical: Prepare a Profit & Loss Statement for an IT firm based on given data. Create a project budget for software development using Tally or Zoho Books. **(11 Hours)**

UNIT III

Basics of Company Laws: Introduction to Company Law: Types of Companies: Private, Public, One-Person Companies. Key Steps in Company Incorporation.

Legal Considerations for IT Businesses: Intellectual Property Rights (IPR): Copyrights, Patents, and Trademarks. Cyber Laws: IT Act 2000 & 2023 and Amendments. Data Protection and Privacy Laws (e.g., GDPR, India's Data Protection Bill).

Corporate Governance: Importance of Ethical Practices in IT. Role of Directors and Compliance Officers.

Contracts and Agreements: Essentials of a Valid Contract. Non-Disclosure Agreements (NDAs), Software Licensing, and Vendor Contracts.

Practical: Draft a simple Non-Disclosure Agreement (NDA) for an IT project. Discuss a real-world case of corporate governance failure in the IT industry. **(10 Hours)**

UNIT IV

Integration and Case Studies: Integration of Finance, Accounting, and Company Laws in IT Project Management: Financial and Legal Planning for IT Startups. Managing Project Costs and Compliance Simultaneously.

Case Studies:

Case Study 1: Funding a tech startup and managing its financial and legal compliance.

Case Study 2: Budget overruns in IT projects and how to address them.

Case Study 3: Cybersecurity breaches and the role of company law.

Practical: Group project: Prepare a financial model, budget, and compliance checklist for a hypothetical IT startup. Presentation of group findings with Q & A. **(09 Hours)**

Assessment:

1. Assignments: 20% (Individual work on case studies and financial exercises)
2. Practical Work: 30% (Group projects, software-based activities)
3. Mid-Term Examination: 20% (Short-answer questions and problem-solving)
4. End-Term Examination: 30% (Comprehensive with case-based questions)

REFERENCE BOOKS:

Accounting

[1] "Financial Accounting for Management" by P. C. Tulsian (A comprehensive guide that explains accounting principles and practices with a focus on management decisions)

[2] "Advanced Accountancy" by R.L. Gupta & M. Radhaswamy (An in-depth book covering advanced accounting topics like partnership accounts, company accounts, and financial statements)

[3] "Financial Accounting" by T.S. Grewal (A well-structured book for understanding the fundamentals of accounting and its various components)

Finance

[4] "Financial Management" by I.M. Pandey (This book provides a detailed explanation of financial management concepts and tools like time value of money, capital budgeting, and risk management)

[5] "Essentials of Financial Management" by Eugene F. Brigham & Joel F. Houston (A solid textbook covering both the theoretical and practical aspects of financial management)

[6] "Corporate Finance" by Ross, Westerfield, and Jaffe (A comprehensive book on corporate finance that covers topics such as capital structure, investments, financial analysis, and valuation)

[7] "Fundamentals of Financial Management" by Prasanna Chandra (Focuses on the key areas of finance, including financial analysis, capital budgeting, and risk management)

Company Laws

[8] "Company Law" by Avtar Singh (A detailed and easy-to-understand guide to the legal aspects of companies, including provisions of the Companies Act, 2013)

[9] "Company Law" by M.C. Kuchhal (This book offers a comprehensive understanding of company law with in-depth discussions on the incorporation, governance, and regulation of companies)

[10] "Business Law" by P.K. Agrawal (Covers various business laws, including company law, contract law, and other legal regulations related to business operations)

[11] "Corporate Laws and Governance" by S. M. Shukla (Focuses on the regulatory and compliance framework surrounding corporate governance, including sections from the Companies Act, 2013.)

IT and Accounting

[12] "Accounting for Managers" by S.N. Maheshwari (This book bridges the gap between accounting and management, with a focus on how accounting principles apply to business management, especially in the IT sector)

[13] "Financial Accounting and Management Accounting" by R.N. Gupta (A good reference for students to understand the practical aspects of accounting and how they relate to managerial decision-making)

Legal Aspects for IT

[14] "Cyber Law in India" by P.K. Sethi (A valuable book for understanding the legal framework of the internet, data privacy, and intellectual property rights in India)

[15] "Information Technology Law: The Law of the Internet, E-Commerce and Digital Communications" by Andrew Murray (A great book for understanding the legal aspects of technology, the internet, and e-commerce in today's digital world)

[16] "Law and the Internet" by D. W. S. (David) Clarke (Focuses on the relationship between law and internet technologies, including key issues in cybersecurity and online data protection)

CA 7625: ENTREPRENEURSHIP AND INNOVATION

Total No. of Lectures: 30	**Total Marks : 50**	**[L – T – P – S]**
No. of Lectures / Week: 2	**Credits : 2**	**[2 – 1 – 0 – 1]**

Course objectives:

Upon successful completion of this course, students should be able to:

- Understand entrepreneurial skillset and mindset
- Identify problems as catalyst for innovation
- Develop customer personas based on initial research

- Enhance customer personas based on market/other feedback
- Examine the core principles of lean startups/design thinking
- Acquire techniques for narrowing down solutions to the most viable options (map problem-JTBD-solution idea)
- Understand contemporary competition and market influences
- Understand solution: prototyping vs. MVPs

Course Outcomes:

After successful completion of this course, the students should:

- Commit to higher career possibilities enabled by entrepreneurial skills
- Scope a problem worth solving with user impact, and personal drivers
- Enhance customer personas based on market/other feedback
- Analyze responses to uncover insights on customer needs/JTBD/problems
- Pivot problem/customer segmentation based on customer insight
- Acquire techniques for narrowing down solutions to the most viable options (map problem-JTBD-solution idea)
- Apply learn method to estimate the Total Addressable Market (TAM)
- Develop & exhibit fundamental skills to prototype
- Apply market analysis in planning
- Craft an initial sales plans

UNIT I

Orientation: Entrepreneurial leadership and its relevance of entrepreneurial mindset for students' career choices-Overview of global/Indian startup ecosystems-Introduction to design thinking/innovation.

Problem Identification: Develop macro industry-problem perspective - Assemble 'real-world' problems - Analyze problems. **(05 Hours)**

UNIT II

Customer Identification: Identify customer segments. Assess 'Jobs-To-Be-Done' (JTBD) for a customer segment-Develop customer personal. **Customer Needs Validation:** Understand customer validation-Validate customer-problem fit-Pivot/refine customer-problem fit for feasibility. **Solution Idea Generation:** Examine ideation techniques-Generate solution ideas. **Opportunity and Competition Mapping:** Identify global competitors-Review industry trends-Estimate market size (TAM). **(06 Hours)**

UNIT III

Prototype Development and MVP: Overview of prototypes & MVP - Build an initial prototype-Prototypes for early validation. **Opportunity (feasibility) assessment:** Map your relative

position in the market-Estimate opportunity size. **Business Modelling:** Examine revenue models-Review and organize the lean canvas-Build the lean canvas for your startup. **(06 Hours)**

UNIT IV

Marketing and Sales Strategy: Understand marketing and sales-Recommend a GTM approach-Map a sales process. **Financial management for profitability:** Understand startup costs-Get started with financial planning-Analyse the path to profitability-Understand bootstrapping strategies & explore funding options. **Team and talent requirements:** Finding co-founders and mentors-Building an initial team. **(07 Hours)**

UNIT V

Venture Idea Pitch Readiness: Scale orientation - Persuasive story-telling for a feasible venture idea. Final Milestone - Venture Idea Feasibility Presentation - **Milestone 3:** Venture idea feasibility (extrnal) jury presentation

Digital marketing: Introduction to digital marketing-Search engine optimization-Search Engine Marketing-Social media marketing-Email marketing-Content marketing

Case Study: eBay thrives in the global marketplace, Boo-Hoo – learning from the largest European dot-com failure, Zopa launches a new lending model, Tesco.com uses the Internet to support its diversification strategy, The new Napster changes the music marketing mix **(06 Hours)**

Text Books

[1] Dr. R. Paulmoni, "Entrepreneurship, Innovation And Creativity", AG Publishing House, First Edition, 2024.

[2] H. James Harrington, "Creativity, Innovation, and Entrepreneurship", Productivity Press; 1st edition (11 December 2018).

[3] Simon Kingsnorth, Digital Marketing Strategy: An Integrated Approach to Online Marketing, Kogan Page Ltd; 3rd edition (31 May 2022)

Reference Books:

[1] Peter F. Drucker, Innovation and Entrepreneurship, Harper Business; Reprint edition (9 May 2006)

[2] Michael Michalko, "Creative Thinkering", New World Library; Original edition (6 September 2011)

SECOND SEMESTER

CA 8125: INFORMATION AND NETWORK SECURITY

Total No. of Lectures : 60 **Total Marks : 100** **[L – T – P – S]**

No. of Lectures / Week : 4 **Credits : 4** **[3 – 1 – 0 – 2]**

Course objectives:

- To provide a foundational understanding of computer networks and security principles.
- To enable students to analyze and apply cryptographic algorithms for secure communication.
- To impart knowledge of network security protocols and mechanisms for securing systems.
- To evaluate and address security challenges in web applications, wireless networks, and emerging technologies.
- To equip students with the ability to implement and manage secure systems using modern tools and industry best practices.

Course Outcomes:

- Understand the fundamental concepts of computer networks and security principles.
- Apply cryptographic techniques to ensure the confidentiality, integrity, and authenticity of data.
- Design and evaluate secure network systems and architectures to safeguard against potential security risks.
- Identify vulnerabilities in networked systems and implement appropriate mitigation strategies.
- Utilize modern network security tools and protocols to effectively address practical security challenges.
- Analyze emerging security trends and their implications on network security.

UNIT I

Fundamentals of Computer Networks and Security: Basics of Computer Networks: OSI and TCP/IP models, types of networks (LAN, WAN, MAN), switching techniques. Network Devices: Routers, switches, firewalls, access points. **Introduction to Network Security:** CIA triad, types of attacks (active, passive), vulnerabilities, threats, and risks. **Security Mechanisms:** Authentication, access control, encryption, firewalls, intrusion detection systems. Overview of Cybersecurity Frameworks: NIST, ISO 27001. **(14 Hours)**

UNIT II

Cryptographic Principles and Algorithms: Introduction to Cryptography: Symmetric and asymmetric cryptography, hash functions. **Classical Encryption Techniques:** Substitution and

transposition ciphers. Symmetric Key Algorithms: DES, AES. **Asymmetric Key Algorithms:** RSA, Diffie-Hellman. Hashing and Digital Signatures: MD5, SHA family, and their applications.

(12 Hours)

UNIT III

Network Security Protocols and Mechanisms: Secure Communication Protocols: SSL/TLS, IPsec, HTTPS. **User Authentication:** Passwords, biometrics, multi-factor authentication. Email Security: S/MIME, PGP. **Wireless Network Security:** WEP, WPA, WPA2. **Security in Emerging Networks:** IoT security challenges and solutions. **(10 Hours)**

UNIT IV

Security in Web Applications and Systems: Web Security Threats: SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF). Secure Web Development Practices: Input validation, secure coding. **Security for Cloud Computing:** Virtualization security, data protection in cloud environments. **Malware and Threats:** Types (viruses, worms, Trojans), anti-malware tools. Incident Response: Logging, monitoring, and recovery procedures. **(12 Hours)**

UNIT V

Emerging Trends and Case Studies in Information Security: Blockchain and Cryptocurrencies: Overview, security aspects. **Ethical Hacking and Penetration Testing:** Tools and methodologies. Case Studies: Analysis of real-world security breaches. Security **Auditing and Risk Management: Tools and best practices. Trends:** AI in security, zero trust architecture, quantum cryptography. **(12 Hours)**

Text Books

[1] William Stallings, "Cryptography and Network Security: Principles and Practice", Pearson Education.

[2] Behrouz A. Forouzan, "Data Communications and Networking", McGraw-Hill.

Reference Books:

[1] AtulKahate, "Cryptography and Network Security", Tata McGraw-Hill.

[2] Michael E. Whitman and Herbert J. Mattord, "Principles of Information Security", Cengage Learning.

[3] Charlie Kaufman, Radia Perlman, and Mike Speciner, "Network Security: Private Communication in a Public World", Pearson Education.

[4] Nina Godbole and SunitBelapure, "Cyber Security: Understanding Cyber Crimes, Computer Forensics, and Legal Perspectives", Wiley India.

[5] Bruce Schneier, "Applied Cryptography", Wiley.

CA 8225: ADVANCED PROGRAMMING WITH JAVA

Total No. of Lectures: 60	**Total Marks: 100**	**[L – T – P – S]**
No. of Lectures/Week: 4	**Credits: 4**	**[3 – 1 – 0 – 2]**

Course Objectives:

- Provide in depth knowledge of Object oriented programming starting from basics.
- Explore the OOP concepts like constructors, method overloading, objects as parameters and nested classes.
- Explore some more advanced OOP concepts like inheritance, interfaces, packages, exception handling, multithreading and generics to understand their use and implementation.

Course outcomes: The candidate will be able to understand

- Understand the Object Oriented Concepts well with real world problems and will be able to model the requirements to solutions.
- Students will be able to build Java applications where they can read from and write to files.
- End of the course student will be able to develop and deliver the power of multithreaded programming and handle the packages well.

UNIT I

Principles of Object-Oriented Programming: A look at procedure-oriented programming – characteristics, Object Oriented programming paradigm, Basic concepts of OOP, Benefits and application of OOP.

Classes, Objects and Methods: Class Fundamentals, Creating Objects – the new operator, Reference Variables and Assignment, Methods, Getters and Setters, Using Parameters – this Keyword.

Arrays and Strings: Arrays in Java, Multidimensional Arrays, Using the Length Member, the For-Each Style for Loop. String class Fundamentals, The String Constructors, String class methods, StringBuffer and StringBuilder classes. **(13 Hours)**

UNIT II

Inheritance: Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, using super under different contexts, Different types of inheritance, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final keyword.

Interfaces: Interface Fundamentals, creating an Interface, implementing an Interface, Implementing Multiple Interfaces. Package Fundamentals, Packages and Member Access, Importing Packages **(12 Hours)**

UNIT III

Exception Handling: Exception Handling Fundamentals, try…catch block, using Multiple catch clauses, throwing an Exception, Throwable class, using throw, throws, finally, creating user defined exceptions and its use.

Multithreaded Programming: Multithreading fundamentals, Thread life cycle, creating a Thread - Thread Class and Runnable Interface, Creating Multiple Threads, Thread Priorities, Synchronization. **(13 Hours)**

UNIT IV

Servlets: MVC Architecture, Introduction to Servlet, Features of Java Servlets, Servlet tasks, Servlet package, Life Cycle of Servlet, Get() and Post() methods, Reading form parameters, Client HTTP Request, Server HTTP Response, Cookies handling, Session Tracking **(10 Hours)**

UNIT V

EJB – EJB architecture and concepts, Responsibilities of EJB Container and Server, Classification of EJB, Anatomy of Session and Entity beans, Differences between CMP and BMP, Annotations, Dependency Injection, Introduction to MVC and Springboot **(12 Hours)**

Text Book:

[1] Herbert Schildt, "Java: The Complete Reference", 8th Edition, Indian Edition, Oracle Press, McGraw- Hill.

[2] Paul Deitel, Harvey Deitel, "Java How to Program", 8th Edition, PHI

[3] Somnath Musib, Spring Boot in Practice, Manning Publications Co., 2022.

Reference books:

[1] Cay Horstman, Gary Cornel, "Core Java: Fundamentals", 8th Edition, 2010, Pearson Asia.

[2] Jana, Debashish, "Java and Object Oriented Programming Paradigm", PHI Learning

[3] Joyce Farrell, "Java For Beginners", Cengage Learning India

[4] Rashmi Kanta Das, "Core Java for Beginners", Vikas Publishing House Pvt. Ltd.

[5] Khalid A. Mughal, Rolf W. Rasmussen, "A Programmer's Guide to Java SCJP Certification", 3rd Edition, Pearson

[6] Walter Savitch, "Java An Introduction to Problem Solving and Programming", Pearson

[7] "Java 7 Programming Black Book", Kogent Learning Solutions Inc., Dreamtech Press.

ELECTIVE I (Theory)

CADE 8325: NATURAL LANGUAGE PROCESSING

Total No. of Lectures: 60	**Total Marks: 100**	**[L – T – P – S]**
No. of Lectures/Week: 4	**Credits: 4**	**[3 – 1 – 0 – 2]**

Course Objectives:

- To understand natural language processing and to learn how to apply basic algorithms in this field.
- To get acquainted with the basic concepts and algorithmic description of the main language levels: morphology, syntax, semantics, and pragmatics.
- To design and implement applications based on natural language processing
- To implement various language Models.
- To design systems that use NLP techniques

Course Outcomes:

The students will:

- Have a broad understanding of the field of natural language processing.
- Have a sense of the capabilities and limitations of current natural language technologies,
- Be able to model linguistic phenomena with a formal grammar.
- Be able to Design, implement, and test algorithms for NLP problems
- Understand the mathematical and linguistic foundations underlying approaches to the various areas of NLP
- Be able to apply NLP techniques to design real-world NLP applications such as machine translation, text categorization, text summarization, and information extraction

UNIT I

Introduction to NLP and Preprocessing: Introduction: History of NLP, Generic NLP system, levels of NLP, Knowledge in language processing, Ambiguity in Natural language, stages in NLP, Applications of NLP. Evaluation of NLP System.

Text Preprocessing: Tokenization, Stemming and Lemmatization, Stopword removal, Text normalization. Regular Expressions and pattern matching for text manipulation. **Speech Processing:** Basics of Speech Recognition and Synthesis. **Basic English Concepts:** Introduction to CFG, Various Grammar-based Language Models, Sentence, Active and Passive voice, Dictionary Features. **(12 Hours)**

UNIT II

Core Techniques for Word and Syntax Analysis: Word Level Analysis: Morphology analysis – survey of English Morphology, Inflectional morphology & Derivational morphology, Lemmatization, Regular expression, finite automata, finite state transducers (FST), Morphological

parsing with FST, Lexicon free FST Porter stemmer. N–Grams-N-gram language model, N-gram for spelling correction.

Syntax Analysis: Part-Of-Speech tagging (POS)- Tag set for English (Penn Treebank), Rule-based POS tagging, Stochastic POS tagging, Issues –Multiple tags & words, Unknown words. Sequence labeling: Hidden Markov Model (HMM), Maximum Entropy, and Conditional Random Field (CRF).

(12 Hours)

UNIT III

Parsing Techniques and Transformer Architectures: Parsing Techniques: Parsing Process, Algorithm, Efficient Parsing, Types of Parsing Techniques, Transition networks, Human Parsing, Feature Grammar, Parsing with Unification constraints, Parsing Integration and Unification, Prolog Grammar Parser Generator.

Advancements in NLP: Introduction to Transformers and Attention Mechanisms. Overview of architectures like BERT, GPT, and their applications **(12 Hours)**

UNIT IV

Semantic Analysis and Word Representation: Semantic Analysis: Lexical Semantics, Attachment for fragment of English- sentences, noun phrases, Verb phrases, prepositional phrases, Relations among lexemes & their senses –Homonymy, Polysemy, Synonymy, Hyponymy, WordNet, Robust Word Sense Disambiguation (WSD) Dictionary based approach Syntax Driven Semantic Analysis Approach.

Word Representation: Word Embeddings: Word2Vec, GloVe, FastText, and Transformer embeddings. **Contextual embeddings:** Comparison with traditional models. **(12 Hours)**

UNIT V

Innovations and Applications in NLP: Knowledge Representation for NLP: Human Language Technology, Information Discovery, issues in Semantic Representation.

Natural Language Generation: Architecture of NLG Systems- Generation Tasks and Representations- Application of NLG.

Machine Translation: Challenges in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches-Statistical and Neural Machine Translation.

Pragmatics: Discourse–reference resolution, reference phenomenon, syntactic & semantic constraints on coreference.

Conversational AI: Building chatbots and dialogue systems using frameworks like Rasa or OpenAI API. **(12 Hours)**

TEXT BOOKS:

[1] Daniel Jurafsky and James H. Martin, “Speech and Language Processing", 2end Ed, 2014, Pearson Education.

[2] James Allen, "Natural Language Understanding", 2nd edition. Bejamin Cummings publishing.

REFERENCE BOOKS:

[1] Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press.

[2] Manning, Christopher and Heinrich, Schutze, "Foundations of Statistical Natural Language Processing", 2016, Create Space Independent Publishing Platform.

[3] Jacob Eisenstein, "Introduction to Natural Language Processing",2019,MIT Press

[4] Carol Genetti, "How Languages Work: An Introduction to Language and Linguistics, 2019, Cambridge University Press.

CADE 8425: IMAGE PROCESSING AND PATTERN RECOGNITION

Total No. of Lectures: 60	**Total Marks: 100**	**[L – T – P – S]**
No. of Lectures/Week: 4	**Credits: 4**	**[3 – 1 – 0 – 2]**

Course objectives:

Upon successful completion of this course, students should be able to:

- Understand the fundamental concepts of digital image processing and mathematical tools for image analysis.
- Learn essential image enhancement, segmentation, and representation techniques.
- Comprehend feature extraction, dimensionality reduction, and classification methods for pattern recognition.
- Explore supervised and unsupervised learning techniques for pattern classification.
- Analyze real-world applications of image processing and pattern recognition.

Course Outcomes:

After you have finished this course, you should have

- Demonstrate a deep understanding of image processing techniques in both spatial and frequency domains.
- Develop skills to apply image enhancement and restoration techniques.
- Analyze and implement segmentation, feature extraction, and representation methods.
- Understand and apply statistical and machine learning approaches to pattern recognition.
- Solve real-world problems using modern image processing and pattern recognition techniques.

UNIT I

Fundamentals of Digital Image Processing: Introduction to Digital Image Processing: Historical development and applications of DIP. Components of an image processing system. **Digital Image Fundamentals:** Visual perception principles and image formation models. Sampling, quantization, and image resolution. Pixel relationships (neighborhoods, connectivity,

adjacency). **Mathematical Preliminaries:** Linear and nonlinear operations. Basics of convolution and correlation.

Tools and Platforms for Image Processing: (MATLAB/OpenCV Basics) **(10 Hours)**

UNIT II

Image Enhancement and Restoration: Image Enhancement in Spatial Domain: Point operations (gray-level transformations). Histogram processing and equalization. Spatial filtering techniques (smoothing and sharpening). **Image Enhancement in Frequency Domain:** Fourier transform and discrete Fourier transform (DFT). Frequency domain filters (low-pass, high-pass, bandpass). **Image Restoration:** Image degradation/restoration process. Noise models (Gaussian, impulse, etc.). Wiener and inverse filtering techniques. (**12 Hours)**

UNIT III

Image Segmentation and Feature Extraction: Image Segmentation Techniques: Edge detection (Sobel, Canny). Thresholding (global, adaptive, Otsu's method). Region-based segmentation (region growing, region splitting/merging). **Feature Extraction and Dimensionality Reduction:** Corner detection (Harris), texture features (GLCM). SIFT, SURF, HOG, and Gabor filters. PCA and LDA for feature reduction. **Image Representation:** Boundary descriptors (perimeter, compactness). Regional descriptors (moments, textures). **(13 Hours)**

UNIT IV

Pattern Recognition Basics: Introduction to Pattern Recognition: Components of a pattern recognition system. Supervised vs. unsupervised learning. Feature space, feature selection, and dimensionality. **Statistical Pattern Recognition:** Parametric classification (Bayes classifier). Non-parametric techniques (k-NN, Parzen window). **Unsupervised Learning Methods:** Clustering techniques (k-means, hierarchical clustering). Evaluation metrics for clustering.

(13 Hours)

UNIT V

Advanced Pattern Recognition and Applications: Fuzzy and Syntactic Pattern Recognition: Fuzzy logic in pattern classification. Fuzzy clustering (FCM). Grammar-based syntactic pattern recognition. **Performance Evaluation:** Risk, error probabilities, and ROC curves. Accuracy, precision, recall, F1 score. **Applications of Pattern Recognition:** Face recognition, object detection, and character recognition. Medical imaging, remote sensing, and biometrics.

(12 Hours)

Text Books

[1] Rafael C. Gonzalez and Richard E. Woods: “Digital Image Processing”, 4th Edition, Pearson, 2018.

[2] Sergios Theodoridis, Konstantinos Koutroumbas: "Pattern Recognition", 5th Edition, Academic Press, 2018.

Reference Books:

[1] Milan Sonka, Vaclav Hlavac, Roger Boyle: "Image Processing, Analysis, and Machine Vision", 4th Edition, Cengage Learning, 2014.

[2] Christopher M. Bishop: "Pattern Recognition and Machine Learning", Springer, 2016.

[3] Richard O. Duda, Peter E. Hart, David G. Stork: "Pattern Classification", 2nd Edition, Wiley, 2012.

[4] Scott E. Umbaugh: "Digital Image Processing and Analysis", 3rd Edition, CRC Press, 2017

ELECTIVE II (Theory)

CADE 8525: BIG DATA ANALYTICS

Total no of lectures: 60	**Total marks: 100**	**[L – T – P –S]**
No of lectures / week: 4	**Credits: 4**	**[3 – 1 – 0 – 2]**

Course Objectives:

- The course aims to cover Big Data Fundamentals, including the characteristics of Big Data, the sources Big Data (such as social media, sensor data, geospatial data etc), as well as the challenges imposed around information management, data analytics, privacy and security, as well as platforms and architectures.
- Emphasis will be given on non-relational databases by examining techniques for storing and processing large volumes of structured and unstructured data, streaming data as well as complex analytics on them.
- Data warehouses will be also presented as a solution to handling big data and business intelligence applications.

Course Outcomes: Upon Completion of the course, the students will be able to

- Identify and distinguish big data analytics applications
- Describe big data analytics tools
- Explain big data analytics techniques
- Present cases involving big data analytics in solving practical problems
- Conduct big data analytics using system tools
- Suggest appropriate solutions to big data analytics problems

UNIT I

Introduction to big data: Data, Characteristics of data and Types of digital data: Sources of data, Working with unstructured data, Evolution and Definition of big data, Characteristics and Need of big data, Challenges of big data.

Big data analytics: Overview of business intelligence, Data science and Analytics, Meaning and Characteristics of big data analytics, Need of big data analytics, Classification of analytics, Challenges to big data analytics, Importance of big data analytics, Basic terminologies in big data environment. **(11 Hours)**

UNIT II

NoSQL Data Management : Introduction to NoSQL, aggregate data models, aggregates, key-value and document data models; relationships, graph databases, schemaless databases, materialized views, distribution models, sharding; master-slave replication; peer-peer replication; sharding and replication; consistency; relaxing consistency; version stamps; Case studies using MongoDB. **(12 Hours)**

UNIT III

Introduction to Hadoop: Introducing Hadoop, need of Hadoop, limitations of RDBMS, RDBMS versus Hadoop, Distributed Computing Challenges, History of Hadoop , Hadoop Overview, Use Case of Hadoop, Hadoop Distributors, HDFS (Hadoop Distributed File System) , Types of files in HDFS- Strengths and alternatives of HDFS, Block System- Read-write Process for data, Processing Data with Hadoop, Managing Resources and Applications with Hadoop YARN (Yet another Resource Negotiator), Interacting with Hadoop Ecosystem **.** **(13 Hours)**

UNIT IV

Introduction to MAPREDUCE Programming: Introduction, Sample MapReduce application: Wordcount- MapReduce Data types and Formats Writing MapReduce Programming-Testing MapReduce Programs- MapReduce Job Execution- Shuffle and Sort Managing Failures- Progress and Status Updates. Mapper, Reducer, Combiner, Partitioner, Searching, Sorting, Compression, Real time applications using MapReduce, Data serialization and working with common serialization formats, Big data serialization formats. **(11 Hours)**

UNIT V

Introduction to Hive: Introduction to Hive, Hive Architecture, Hive Data Types, Hive File Format, Hive Query Language (HQL), User-Defined Function (UDF) in Hive. Hive Architecture-Components- Data Definition- Partitioning- Data Manipulation- Joins, Views and Indexes,Hive Execution-

Introduction to Pig: Introduction to Pig, Pig Architecture- Pig Latin Data Model- Latin Operators-Loading Data- Diagnostic Operators the Anatomy of Pig, Pig on Hadoop, Pig Philosophy, Use Case for Pig: ETL Processing, Group Operators- Pig Joins- Row Level Operators- Pig Built-in function-User-defined functions- Pig Scripts. Data Types in Pig, Running Pig, Execution Modes of Pig, HDFS

Commands, Relational Operators, Piggy Bank, Word Count Example using Pig, Pig at Yahoo!, Pig versus Hive **(13 Hours)**

Text Books:

[1] Seema Acharya, Subhashini Chellappan "Big Data and Analytics", 1st Edition, 2015, Wiley International

[2] S. Mohanthy, Madhu Jagadish, Harsh Srivatsa, "Big Data Imperatives: Enterprise Big Data Warehouse, BI Implementations and Analytics", 1st Edition, 2015, Wiley Apress

Reference Books:

[1] Jey Liebowitz, "Big Data and Business Analytics", 1st Edition, 2014, CRC Press

[2] Tom White, "Hadoop – Definitive Guide", 1st Edition, 2015, O'Reilly Publishers

[3] Boris Lublinsky, Kevin t. Smith, Alexey Yakubovich, "Professional Hadoop Solutions",1st Edition, 2015, Wiley India Publications.

[4] E Caprilo, Dean Wampler, "Hive Programming", 1st Edition, 2015, O'Reilly Publishers

[5] [Sherif Sakr, "Large Scale and Big Data: Processing and Management", 2014, CRC Press.

CADE 8625: DATA SCIENCE

Total no of lectures: 60	**Total marks: 100**	**[L – T – P – S]**
No of lectures / week: 4	**Credits: 4**	**[3 – 1 – 0 – 2]**

Course Objectives: The objectives of this course are to provide with

- data sampling/cleaning in order to get an informative, manageable data set
- data storage and management in order to be able to access data.
- exploratory data analysis to generate hypotheses and intuition about the data.
- prediction based on statistical tools such as regression, classification, and clustering.
- communication of results through visualization, stories, and interpretable summaries

Course Outcomes: After successful completion of this course, you will be able to…

- Use data management techniques to store data
- Use statistical methods and visualization to quickly explore data
- Apply statistics and computational analysis to make predictions based on data
- Implement data-intensive computations on cluster and cloud infrastructures.
- Effectively communicate the outcome of data analysis using descriptive statistics and visualizations

UNIT I

Data Science: Data science process – roles, stages in data science project; working with data from files; working with relational databases; exploring data, managing data, cleaning and sampling for modeling and validation, introduction to NoSQL.

Data Analytics Life Cycle: Introduction to Big data Business Analytics; State of the practice in analytics role of data scientists; Key roles for successful analytic project; Main phases of life cycle - Developing core deliverables for stakeholders. **(10 Hours)**

UNIT II

Data Mining: Introduction -Data mining task. Data Preprocessing**:** Descriptive Data Summarization - Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization. Data visualization

Association Rule Mining: Basic concepts – Market basket analysis Scalable Frequent Itemset Mining Methods – Improving efficiency – Mining various kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-based Association Mining. **(10 Hours)**

UNIT III

Feature selection: Introduction-Steps in feature selection-Principal-component analysis for feature reduction.

Ensemble learning: Introduction-Ensemble-learning framework-Supervised ensemble learning-Unsupervised ensemble learning-Semi-supervised ensemble learning-Issues and challenges.

Machine learning: Introduction-Machine Learning paradigms-Inductive bias-Evaluating a classifier **(12 Hours)**

UNIT IV

Regression: Introduction-Regression-Evaluating linear regression-Multidimensional linear regression-Polynomial regression-Overfitting in regression-**Reducing overfitting in regression:** regularization

Classification: Basic concepts – Decision tree induction - Bayes classification methods – Rule based classification – Support Vector Machine.

Predictive Analytics: Simple linear regression: Coefficient of determination, Significance tests, Residual analysis, Confidence and Prediction intervals;

Multiple linear regression: Coefficient of determination, Interpretation of regression coefficients, Categorical variables, outliers, Regression Model Building;

Logistic and Multinomial Regression: Logistic function, Estimation of probability using logistic regression. **(15 Hours)**

UNIT V

Data Science in practice: Need of Data Science in the real world, Hands-on Data Science with Python, Dataset preprocessing, Feature selection and normalization, Classification, Clustering

Hypothesis testing: Null and alternate hypotheses; Types of errors, Level of significance, Power of a test, ANOVA; Test for goodness of fit

Forecasting: Moving average, Exponential smoothing, Casual Models.

Application of predictive analytics in retail, direct marketing, health care, financial services, insurance, supply chain, etc. **(13 Hours)**

Text Books:

[1] Jugal K. Kalita, Dhruba K. Bhattacharyya, Swarup Roy, Fundamentals of Data Science: Theory and Practice, Academic Press-Elsevier, 2024.

[2] Nina Zumel, John Mount, "Practical Data Science with R", Manning Publications, 2014.

[3] Jain Pei, Jiawei Han, Micheline Kamber, "Data Mining : Concepts and Techniques", 3rd Ed, 2011, Elsevier

[4] Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning", 2nd Edition, 2011, Springer Series in Statistics.

[5] Daniel T. Larose, Chantal D. Larose "Data Mining and Predictive Analytics", 2nd Ed, 2016, John Wiley & Sons Inc.,

Reference Books:

[1] Ken Black, "Applied BUSINESS STATISTICS Making better business decisions", 7th Ed, 2016, John Wiley & Sons Inc.,

[2] R N Prasad, Seema Acharya, "Fundamentals of Business Analytics", 1st Ed, 2011, Wiley India

[3] Tony Ojeda, Sean Patrick Murphy, Benjamin Bengfort, Abhijit Dasgupta, "Practical Data Science Cookbook", 1st Ed, 2014, Packt Publishing Ltd..

[4] Nathan Yau, "Visualize This: The Flowing Data Guide to Design, Visualization", 2011, Wiley

[5] Daniel T. Larose, "Discovering Knowledge in Data: An Introduction to Data Mining", 2011, Wiley Intl

[6] Thomas W Miller, "Modeling Techniques in Predictive Analytics", 1st Ed, 2013, Pearson

[7] Peter Adrians, Rolf Zantinge, "Data Mining", 1st Edition, 2010, Pearson Education

[8] Vikram Pudi, Radhakrishna, "Data Mining", 2nd Edition, 2011, Oxford University Press.

[9] Richard Roiger, Michael Getz, "Data Mining : A Practical Based Primer", 1st Ed, 2010, Pearson

[10] Arun K Pujari, "Data Mining Techniques", 2nd Ed, 2013, University Press

CA 8P125: ADVANCED JAVA LAB

1. Write a java program that demonstrates the creation and implementation of various Arraylist methods. Traverse through the list before and after each operation.
2. Write a java program that demonstrates the use of a linked list by performing Linkedlist specific operations.
3. Write a program in java to manage a set of unique student ids for a university. You need to ensure that no duplicate ids are present and perform various operations on this set of student ids using java Hashset class.
4. Write a java program to use a function retrieveCourseFromDatabase() to take courseName, courseID, courseCategory as inputs. Use get() and set() to store and retrieve data. Display using printCourseDetails(string courseName,string coursed,string courseCategory). Update the courseName to "Computing in java".
5. Write a program in java to implement Model View Controller architecture used in J2EE

6. Write a program in java to demonstrate the various stages of a servlet lifecycle
7. Create a Java Server Page that displays a form to input the username and age. When submitted, the page will display the entered information and provide the current date and time using jsp elements.
8. Create a Java Server Page to display a form. When submitted, handle the request using both GET and POST methods. The form will accept the username and based on the request method, the page will display a message.
9. Write a program to handle session using cookies. It should print the username and email and implement various methods associated with it.
10. Create a Java program that connects to a MySQL database to manage a simple "student" records system. You will use the createStatement() method to execute SQL statements.
    a. Create a Database: Create a database named school_db.
    b. Create a Table: Within the school_db, create a table named students with the following structure:
        (i) student_id (INT, Primary Key, Auto Increment)
        (ii) name (VARCHAR(100))
        (iii) age (INT)
        (iv) grade (VARCHAR(10))
    c. Insert Records: Use the createStatement() method to insert at least three student records into the studentstable.
    d. Retrieve Records: Use the same method to retrieve and display all records from the students table.
    e. Drop the Table: Write a statement to drop the students table to clean up.
11. Create a Java program that connects to a MySQL database
    a. Create a Database
    b. Create a Table: Table named "books" with the following structure:
        i. book_id (INTEGER, Primary Key, Auto Increment)
        ii. title (TEXT)
        iii. author (TEXT)
        iv. published_year (INTEGER)
    c. Insert Records: Implement a method that uses PreparedStatement to insert new book records. Prompt the user for the book's title, author, and published year.
    d. Retrieve Records: Implement a method to retrieve and display all books in the "books" table.
    e. Update Records: Implement a method to update the details of a book by its book_id. The user should be prompted for the book_id and the new details.
    f. Delete Records: Implement a method to delete a book record based on its book_id.
12. Create a Java program that connects to a MySQL database and performs operations on an "employees" table using CallableStatement.
    a. Create a Database
    b. Write stored procedures in SQL to insert and retrieve records
    c. Create a Table: Table named employees with
        (i) emp_id (INTEGER, Primary Key, Auto Increment)
        (ii) name (TEXT)
        (iii) position (TEXT)
        (iv) salary (DECIMAL)

d. Insert Records
e. Retrieve Records

13. Create a Java program that connects to a MySQL database and performs CRUD operations on an "employees" table using CallableStatement.
    a. Create a Database
    b. Write stored procedures in SQL to Update and Delete records
    c. Create a Table: Table named employees with
        (i) emp_id (INTEGER, Primary Key, Auto Increment)
        (ii) name (TEXT)
        (iii) position (TEXT)
        (iv) salary (DECIMAL)
    d. Update Records
    e. Delete Records

ELECTIVE I (Practical)

CADE 8P325: NATURAL LANGUAGE PROCESSING LAB

1. Implement text tokenization, stemming, lemmatization, and stopword removal using Python libraries such as NLTK or SpaCy.
2. Perform text normalization and regular expression-based pattern matching for text manipulation.
3. Implement a menu driven program for basic speech-to-text and text to speech using libraries like Google Speech Recognition API or SpeechRecognition in Python.
4. Create a menu driven program to convert sentences between active and passive voice automatically.
5. Develop a grammar correction tool that identifies and corrects grammatical errors in English sentences.
6. Implement morphological parsing using finite state transducers (FST).
7. Develop a lexicon-free morphological analyzer using the Porter stemmer algorithm.
8. Use the N-gram model for spelling correction and word prediction.
9. Implement a rule-based POS tagging system for English text.
10. Develop a stochastic POS tagging system using Hidden Markov Models (HMM).
11. Create a program to construct and visualize a context-free grammar (CFG) for simple English sentences.
12. Design a program to perform Word Sense Disambiguation (WSD) using dictionary-based approaches.
13. Build a basic Natural Language Generation (NLG) system for generating simple sentences.
14. Develop an application to translate text from English to another language using rule-based or statistical machine translation techniques.
15. Implement extractive summarization techniques using Python libraries.

16. Project Work: Apply multiple NLP techniques to solve a real-world problem.

CADE 8P425: IMAGE PROCESSING LAB

1. **Image Representation and Basic Operations:** Read, display, and save images; perform pixel-level operations, resizing, rotation, and transformations using Python.
2. **Gray-Level Transformations:** Implement contrast stretching, logarithmic, power-law, and negative transformations on images.
3. **Image Quantization:** Simulate image quantization by reducing bits per pixel and analyze visual degradation using Python.
4. **Spatial Domain Filtering:** Apply smoothing filters (mean, median, Gaussian) and sharpening filters (Laplacian, high-pass) for noise reduction and enhancement.
5. **Frequency Domain Filtering:** Perform Fourier Transform and inverse Fourier Transform; apply Homomorphic filtering to correct uneven lighting and enhance contrast.
6. **Image Restoration:** Simulate image degradation (motion blur, Gaussian noise) and restore images using Wiener filtering and inverse filtering techniques.
7. **Edge Detection:** Implement edge detection algorithms like Sobel, Prewitt, Roberts, and Canny.
8. **Thresholding for Image Segmentation:** Perform global thresholding (Otsu's method) and adaptive thresholding for segmenting images.
9. **Region-Based Segmentation:** Implement region-growing and splitting-merging algorithms for image segmentation.
10. **Feature Extraction:** Extract features using GLCM and SIFT methods with OpenCV.
11. **Classification Techniques:** Implement K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) for image classification and evaluation performance.
12. **Bayesian Classifier and Performance Evaluation:** Implement a Bayesian classifier and evaluate using accuracy, precision, recall, F1-score, and confusion matrix.
13. **Clustering Algorithms:** Apply unsupervised clustering algorithms like K-Means and hierarchical clustering for segmenting image data.
14. **Applications of Pattern Recognition:** Implement real-world applications like OCR, face recognition, or object detection and localization using Haar cascades or pre-trained models.
15. **Fuzzy Logic for Pattern Recognition:** Implement fuzzy classification where objects can belong to multiple classes with varying degrees of membership.
16. **Noise Removal and Image Enhancement:** Apply noise removal techniques like Gaussian, median, and bilateral filters to enhance image quality.

ELECTIVE II (Practical)

CADE 8P525: BIG DATA ANALYTICS LAB

1. Write a program to categorize and visualize different types of digital data (structured, unstructured, semi-structured) using Python.
2. Write a Python script to process unstructured text data (e.g., a large log file) and count word frequencies using libraries like NLTK.
3. Implement a Python program to classify given datasets (e.g., sales, weather, or social media) into descriptive, diagnostic, predictive, and prescriptive analytics using NumPy and Pandas.
4. Create a MongoDB database to store student information, including name, age, courses, and grades. Perform basic CRUD operations.
5. Implement an application to store product details (e.g., ID, name, price, and category) using key-value pairs in MongoDB.
6. Use a graph database (like Neo4j or MongoDB with relationships) to model a social network where nodes represent users and edges represent relationships (e.g., friends or followers).
7. Perform basic file operations (upload, download, and delete) in Hadoop Distributed File System (HDFS) using the Hadoop CLI.
8. Write a program to read data from a local file, process it, and store it in HDFS using Python or Java.
9. Implement a basic MapReduce program in Java or Python to count the occurrences of words in a large text file.
10. Develop a MapReduce job to sort a large dataset (e.g., sorting product prices or names).
11. Implement a custom combiner and partitioner in MapReduce to improve the performance of a data aggregation job (e.g., calculating total sales by region).
12. Create a Hive table with a sample dataset (e.g., employee data) and perform queries to filter, group, and aggregate data using Hive Query Language (HQL).
13. Use Hive to partition a dataset (e.g., sales data by region and date) and query specific partitions. Demonstrate the performance difference with and without indexing.
14. Write a Pig script to load a large dataset (e.g., weather data), perform ETL operations, and calculate average values (e.g., temperature or rainfall) using group operators and built-in functions.

CADE 8P625: DATA SCIENCE LAB

1. Implementation of Descriptive and Inferential statistics on any dataset from the popular data sources.
2. Implementation of data preprocessing techniques such as data cleaning, data reduction, data transformation, normalization, and data integration on any dataset from the UCI repository.
3. Implement the various regression techniques on any appropriate dataset from the popular data repositories, evaluate them by their assessment metrics and visualise the evaluation results as graphs.

4. Implement the various classification techniques on any appropriate dataset from the popular data repositories, evaluate them by their assessment metrics, and visualise the evaluation results as graphs.
5. Implement the various Hypothesis testing on any appropriate dataset from the popular data repositories.
6. Implement the various Feature Selection techniques on any appropriate dataset from the popular data repositories.
7. Implement the various cluster analysis techniques for any appropriate dataset from the popular data repositories.
8. Implement the various Association-rule mining techniques on any appropriate dataset from the popular data repositories, evaluate them by their assessment metrics, and visualize the evaluation results as graphs.
9. Use the diabetes data set from UCI and Pima Indians Diabetes data set for performing the following:
    (i) Univariate analysis: Frequency, Mean, Median, Mode, Variance, Standard Deviation, Skewness and Kurtosis.
    (ii) Bivariate analysis: Linear and logistic regression modeling
    (iii) Multiple Regression analysis
    (iv) Also compare the results of the above analysis for the two data sets
10. Apply and explore various plotting functions on UCI data sets.
    (i) Normal curves
    (ii) Density and contour plots
    (iii) Correlation and scatter plots
    (iv) Histograms
    (v) Three-dimensional plotting

CA 8725: INTERNET OF THINGS

Total no of lectures: 42	**Total marks: 100**	**[L – T – P – S]**
No of lectures / week: 3	**Credits: 3**	**[2 – 1 – 0 – 0]**

Course Objectives:

- This subject provides a broad study of various latest internet technologies and underlying communication technologies used in the Internet of Things (IoT).
- The main emphasis is to combine technical concepts with a high-level understanding of IoT networks and communication media used in the IoT.

- The subject provides a comprehensive understanding of the Internet of Things and explores topics such as IoT architecture, IoT communication protocols, security in IoT and IoT applications in the modern world.
- The subject also examines business use-cases of IoT in real-world environments such as industrial and home automation.

Course Outcomes:

Upon successful completion of this course, the participant will be able to:

- The knowledge and understanding of the security and ethical issues of the Internet of Things.
- Conceptually identify vulnerabilities, including recent attacks, involving the Internet of Things.
- Conceptually describe countermeasures for Internet of Things devices.
- Analyse the societal impact of IoT security events; Develop critical thinking skills.
- Compare and contrast the threat environment based on industry and/or device type.

UNIT I

The IoT Networking Core : Technologies involved in IoT Development: Internet/Web and Networking Basics OSI Model, Data transfer referred with OSI Model, IP Addressing, Point to Point Data transfer, Point to Multi Point Data transfer & Network Topologies, Sub-netting, Network Topologies referred with Web, Introduction to Web Servers , IoT Communication APIs, IoT Enabling Technologies- Wireless Sensor Networks, Cloud Computing, Big Data Analytics, Communication Protocols , Embedded Systems, IoT Levels & Deployment Templates. **(9 Hours)**

UNIT II

IoT and M2M: Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT, Software Defined Networking, Network Function Virtualization, IoT Platform Design Methodology, Introduction, IoT Design Methodology, Step1: Purpose and requirement specification, Step2: Process Specification, Step 3: Domain Model Specification, Step 4: Information Model Specification, Step 5: Service Specification, Step 6: IoT Level Specification, Step 7: Function View Specification, Step 8: Operational View Specification, Step 9: Device and Component Integration, Step 10: Application Development, IoT System **(9 Hours)**

UNIT III

IOT Data Link Layer and Protocols: PHY/MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), Wireless HART, Z-Wave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7 - Network Layer-IPv4, IPv6, 6LoWPAN, 6TiSCH, ND, DHCP, ICMP, RPL, CORPL, CARP Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP)- (TLS, DTLS) – Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT **(8 Hours)**

UNIT IV

User Experience Design and the Technology of Connected devices: UX for IoT Different, Design model for IoT. Logical Design-Functions, Modules, Packages, File Handling, Date Time applications, Classes, Python Packages of Interest for IoT. IoT Physical Devices and End Points: What is an IoT Device, Exemplary Device Raspberry Pi, About the Board, Linux on Raspberry Pi, Raspberry pi interfaces, programming raspberry pi with python, other IoT devices. Bridging

physical and digital: sensors and actuators, the challenge of powering devices, conserving battery life **(9 Hours)**

UNIT V

Case Study & advanced IoT Applications: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT electronic equipments. Use of Big Data and Visualization in IoT, Industry 4.0 concepts. Sensors and sensor Node and interfacing using any Embedded target boards. Various IoT platforms.

SELF-LEARNING COMPONENT: Various sensors available in market – application of various sensor – Their specifications – code used to connect these sensors into Microcontroller board – Various microcontroller boards available in market – Arduino IDE download – usage of this IDE to carryout projects. **(7 Hours)**

Text Books:

[1] Arshdeep Bahga, Vijay audisetti, Internet of Things," A Hands on Approach", University Press, 2014. 2.

[2] Michael Millen, "The Internet of Things", Pearson, 2015.

[3] Adrian McEwen & Hakim Cassimally, "Designing the Internet of Things", ISBN 978-81-265-5686-1 Wiley Publication, 2013

[4] Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", 1st Edition, 2016, Willy Publications

SWAYAM/NPTEL/MOOCS:

[1] https://www.udemy.com/ Internet of Things

[2] https://www.coursera.org/learn/ Internet of Things

[3] https://nptel.ac.in/courses/106106133/