

Scheme and Syllabus of
Master of Computer Applications

(With Elective Specialisation in Applied AI/ Data Science)

(02-Year Post Graduate Program)

Batch 2026- 27 onwards

By

Skill Department of Computer Science/ IT
Under
Skill Faculty of Engineering and Technology



Shri Vishwakarma Skill University
Dudhola, Palwal-121102, Haryana

Developed By

Dr. Monika Jangra

(Skill Assistant Professor)
Skill Department of CS/IT

Dr. Anu Chaudhary

(Skill Assistant Professor)
Skill Department of CS/IT

Ms. Madhuri Saini

(Skill Instructor)
Skill Department of CS/IT

Verified By:

Prof. (Dr) Usha Batra

Chairperson
Skill Department of CS/IT

Prof. (Dr) Aashish Shrivastava

Dean
Skill Faculty of Engineering and Technology,
Shri Vishwakarma Skill University, Palwal

Prof. (Dr) Vikram Singh

Dean Academics
Shri Vishwakarma Skill University, Palwal

1.1 VISION and MISSION

Vision of the University

To emerge as one of the foremost institutions of quality in skill education acknowledges by industry, nationally and internationally.

Mission of the University

To focus on quality of skill education, aspired by students and admired by industry, nationally and internationally.

Vision of the Department

To build highly skilled and competent professionals with a futuristic outlook for research and innovation who can also generate sustainable employment with virtuous professional values.

Mission of the Department

1. To impart skill education and training as per industry needs in the domain of Computer Science & Engineering.
2. To develop and maintain the state of art infrastructure for building sustainable skilled IT ecosystem in the domain.
3. To cultivate the environment for fostering new ideas, research and innovations leading to emergence of skilled entrepreneurs, leaders, innovators and professionals.

1.2 PROGRAM OUTCOME

This program is aimed at skilling the students for the jobs in IT industry at and aims at building the following key competencies amongst the learner.

PO1: Computational Knowledge

Apply knowledge of computing fundamentals, computing specialisation, mathematics, and domain knowledge appropriate for the computing specialisation to the abstraction and conceptualisation of computing models from defined problems and requirements.

PO2: Problem Analysis

Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PO3: Design / Development of Solutions

Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

PO4: Conduct Investigations of Complex Computing Problems

Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern Tool Usage

Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

PO6: Professional Ethics

Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.

PO7: Life-long Learning

Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.

PO8: Project Management and Finance

Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO9: Communication Efficacy

Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

PO10: Societal and Environmental Concern

Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.

PO11: Individual and Team Work

Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

PO12: Innovation and Entrepreneurship

Identify a timely opportunity and use innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

1.3 PROGRAM SPECIFIC OUTCOMES

PSO1: Graduates will be able to apply analytical and logical reasoning, along with programming knowledge, to design and develop efficient software solutions for real-world problems.

PSO2: Graduates will demonstrate the ability to apply theoretical and practical knowledge of computer applications to build innovative software products and explore entrepreneurial ventures.

PSO3: Graduates will possess effective communication skills, enabling them to convey technical and non-technical information clearly through oral, written, and digital formats.

PSO4: Graduates will be capable of working effectively in structured environments, demonstrating responsibility and ethical behavior while collaborating within defined roles and boundaries.

1.4 PROGRAM EDUCATIONAL OBJECTIVES

PEO1: Graduates will be equipped with a strong foundation in computer science and applications, combined with professional and entrepreneurial skills, to enable successful careers in the IT industry, pursue higher education, or establish their own ventures.

PEO2: Graduates will demonstrate the ability to work collaboratively in multidisciplinary teams, contribute to software design and implementation, communicate effectively in professional settings, and assist in project documentation and reporting.

PEO3: Graduates will recognize the importance of ethical responsibilities, environmental sustainability, and societal impact, and practice their profession with integrity and awareness of global challenges.

The PO and PSO mapping, PO and PEO mapping are given in table 3 and table 4.

The mapping is defined such as 0: no mapping; 1: Low mapping; 2: Average mapping; 3: Complete mapping.

Table 3. PO and PSO mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
PSO1	2	0	3	0	1	0	0	0	0	0	0	0
PSO2	1	2	0	0	3	0	0	0	0	0	3	0
PSO3	0	0	0	2	0	0	0	0	1	3	0	0
PSO4	0	0	0	0	0	2	1	1	3	0	0	2

Table 4. PO and PEO mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
PEO1	3	1	3	1	2	0	0	0	0	0	0	0
PEO2	0	0	0	0	0	2	0	3	3	3	0	1
PEO3	0	2	0	0	0	1	1	0	0	0	0	0

1.5 DOMAIN WISE CREDIT BREAKUP

Sr. No.	Category	SVSU MCA Program Credits
1	Program Core Courses (PC)	39
2	¹ Ability Enhancement Course (AEC) [courses related to communication and professional ethics, aptitude, leadership, wellness, innovation, digital skills, civic awareness, and sustainable thinking, etc.]	04
3	² Value Added Course (VAC)	04
4	Program Elective (PE)	10
5	Project work, seminar , MOOC and internship in industry or elsewhere	25
6	³ Audit Courses (AU) [courses related to Environmental Sciences, Induction training, Indian Constitution, Essence of Indian Traditional Knowledge, Yoga and Health etc.]	(non-credit)
	Total	82
Note: *The MOOC subject can be qualified at any time during the duration of the Programme(that is from semester first to four)through Swayam platform(UGC) and duration of the course should be equal or more than 12 week ^{1,2,3}Ability Enhancement Courses and Audit Courses may be decided by the time offering. The credits may vary slightly, given that minimum graduation credit criteria fulfil. Bridge courses offered as appropriate for non computer backgrounds students		

1.6 Bridge Course: As per AICTE guidelines the university has to conduct bridge course for non-computer background students having qualifications like B.A, B.com etc. The duration of the bridge course will be of 3 weeks and the following subjects will be taught:

Bridge Course Subjects (Mandatory to Qualify)

Sr. No	Code	Name the Subject	No. of contact hours/week	Total Marks (Internal Only) 30(Exam)+20(Viva)
1	26PMCA53	Fundamentals of Computer System	4	50
2	26PMCA54	Introduction to Database System	4	50
3	26PMCA55	Basics of Data Structure	4	50
4	26PMCA56	Introduction to Database System and Basics of Data Structure Lab	4	50

Note: All the subjects mentioned above for bridge course will be of qualifying nature and there will be no credit for the same .

1.7 SEMESTER WISE SCHEME

The semester wise scheme is given as

SEMESTER I

Subject code	Subject Title	hours per week			Credit	Marks		Category
		L	T	P		I	E	
24PENG01	Professional Communication	1	0	0	1	15	35	AEC
24PENG02	Professional Communication Practical	0	0	2	1	35	15	AEC
26PMCA01	Mathematical Foundations of Computer Applications	3	1	0	4	30	70	PC
26PMCA02	Data Mining and Applications	3	0	0	3	30	70	PC
26PMCA03	Data Mining and Applications Lab	0	0	2	1	35	15	PC
26PMCA04	Programming with C	3	0	0	3	30	70	PC
26PMCA05	Programming with C Lab	0	0	2	1	35	15	PC
26PMCA06	Fundamentals of Data Science	3	1	0	4	30	70	PC
24UYHS01	Yoga and Health	2	0	0	0	30	70	AU
		15	2	6	18			
Total hours		23						

SEMESTER II

Subject code	Subject Title	hours per week				Marks		Category
		L	T	P	Credit	I	E	
26PMCA07	Advanced Data Structures	3	1	0	4	30	70	PC
26PMCA08	Advanced Data Structures Lab	0	0	2	1	35	15	PC
26PMCA09	Object Oriented Programming using Python	3	0	0	3	30	70	PC
26PMCA10	Object Oriented Programming using Python Lab	0	0	2	1	35	15	PC
26PMCA11	Distributed Systems	3	1	0	4	30	70	PC
26PMCA12	Operating System	3	0	0	3	30	70	PC
26PMCA13	Operating System Lab	0	0	2	1	35	15	PC
26PMCA14	Cyber Security and Information Assurance	3	0	0	3	30	70	PC
26PMCA15	Agentic AI	1	0	0	1	30	70	VAC
26PMCA16	Agentic AI Lab	0	0	2	1	35	15	VAC
24UENG03	Employability Skills	1	0	0	1	15	35	AEC
24UENG04	Employability Skills Practical	0	0	2	1	35	15	AEC
		17	2	10	24			
Total hours		29						
After Sem II, student will undergo internship that will be evaluated in Sem III.								

SEMESTER III

Subject code	Subject Title	hours per week			Credit	Marks		Category
		L	T	P		I	E	
26PMCA17-22	Program Elective I	3	1	0	4	30	70	PE
26PMCA23-28	Program Elective I Lab	0	0	2	1	35	15	PE
26PMCA29-36	Program Elective II	3	1	0	4	30	70	PE
26PMCA37-44	Program Elective II Lab	0	0	2	1	35	15	PE
26PMCA45	Software Testing and Quality Assurance	3	0	0	3	30	70	PC
26PMCA46	Excel & Power BI for Data Visualization	1	0	0	1	30	70	VAC
26PMCA47	Excel & Power BI for Data Visualization Lab	0	0	2	1	35	15	VAC
26PMCA48	Capstone Project	0	0	10	5	35	15	Project
24UUI01	Understanding India Skills	2	0	0	0	30	70	AU
26PMCA49	Internship (credits from sem II)	0	0	0	2	35	15	Project
		12	2	16	22			
	Total hours	30						
*Program elective I and II to be selected from List of Electives, may be decided at the time offering as appropriate.								

SEMESTER IV

Subject code	Subject Title	hours per week			Credit	Marks		Category
		L	T	P		I	E	
26PMCA50	Seminar	0	2	0	2	35	15	PC
26PMCA51	Major Project/Dissertation	0	0	24	12	350	150	PC
26PMCA52	Open Elective MOOC	4	0	0	4	30	70	OE
		4	2	24	18			
	Total hours	30						
*The MOOC subject can be qualified at any time during the duration of the Programme(that is from semester first to four)through Swayam platform(UGC) and duration of the course should be equal or more than 12 week								

1.8 LIST OF ELECTIVES I*

Course Code	List of Program Elective I in Applied AI	Course Code	List of Program Elective I in Data Science
26PMCA17	Machine Learning	26PMCA20	Information Retrieval
26PMCA18	Artificial Intelligence	26PMCA21	Machine Learning and Pattern Recognition
26PMCA19	Soft Computing	26PMCA22	Data Analytics
26PMCA23	Machine Learning Lab	26PMCA26	Information Retrieval Lab
26PMCA24	Artificial Intelligence Lab	26PMCA27	Machine Learning and Pattern Recognition Lab
26PMCA25	Soft Computing Lab	26PMCA28	Data Analytics Lab

1.9 LIST OF ELECTIVES II*

Course Code	List of Program Elective II in Applied AI	Course Code	List of Program Elective II in Data Science
26PMCA29	Generative and Explainable AI	26PMCA33	Risk Analysis and Management
26PMCA30	Explainable AI	26PMCA34	Intrusion Detection Systems
26PMCA31	Adversarial AI	26PMCA35	Business Intelligence Systems
26PMCA32	Natural Language Processing	26PMCA36	Big Data Analytics
26PMCA37	Generative and Explainable AI Lab	26PMCA41	Risk Analysis and Management Lab
26PMCA38	Explainable AI Lab	26PMCA42	Intrusion Detection Systems Lab
26PMCA39	Adversarial AI Lab	26PMCA43	Business Intelligence Systems Lab
26PMCA40	Natural Language Processing Lab	26PMCA44	Big Data Analytics Lab

*More elective courses may also be added to the list of open electives and Program Elective (Domain Specific) at the time of offering of courses by the Department.

1.10 Evaluation Scheme of MCA Program

Type of Exam	Internal	External	Total
Theory Subject	30	70	100
Practical Subject	35	15	50
Major Project	350	150	500

Division of Theory Internal Marks – 30 Marks

Sessional 1	Sessional 2	Class Attendance	Assignment/Class Performance/Presentation	Total
5	5	5	15	30

Division of Practical Internal Marks – 35 Marks

Class Performance (Continuous Assessment)	Lab File	Lab Attendance	Internal Viva-Voice	Total
10	10	5	10	35

Division of Practical External Marks – 15 Marks

Performance in Lab during Exam	File/Written Work	External Viva-Voice	Total
5	5	5	15

Division of OJT/In House Training/Project/Seminar Internal Marks – 35

Mid Term Viva	Internal Mid Term Project/Seminar Report	Internal End Term Viva	Presentation	Final Project/Seminar Report	Total
5	10	5	5	10	35

Division of OJT/In House Training/Project/Seminar External Marks – 15

Final Project Report	External Viva-Voice	Total
5	10	15

Division of Major Project Internal Marks –

Mid Term Viva	Internal Mid Term Project/Seminar Report/Synopsis	Internal End Term Viva	Presentation	Final Project/Seminar Report	Total
60	60	60	70	100	350

Division of Major Project External Marks

Report	Presentation	External Viva-Voice	Total
30	30	90	150

MCA Semester I Detailed Syllabus

Professional Communication

Course Code: 24PENG01

Course Credit: 01(1-0-0)
Max. Marks: 50(15I+35E)

Course Objectives:

The course is designed to groom communication skills of the learners to enable them to communicate effectively in diverse professional situations.

Learning Outcomes: After completing this course, the learners will be able to

1. Communicate in diverse professional and inter-cultural situations.
2. Draft various types of messages.
3. Perform well in interviews and meetings.
4. Comprehend and practice communication ethics

Unit	Topics
I Business Communication	Introduction to Business Communication, Types of Communication, 7Cs of Communication, Listening Skills, Legal and Ethical issues in Communication
II Written Communication	Improving personal writing skills, Planning and executing different types of messages, Writing reports, Business Correspondence-letters, Agenda and Minutes, Resume writing
III Interpersonal Skills	Team Intelligence, Assertive Behaviour, Emotional Intelligence, Negotiation Skills, Mentoring and Appraisals, Presentation Skills
IV Inter-Cultural Communication	Culture, Types of Culture, Communication Strategies for cross-cultural communication, Business Etiquettes across cultures
V Interviews	Interviews and its types, Group Discussions, Mock Interviews, Communication during Exit Interviews, Preparing for interviews, Networking and Social media

Recommended Readings:

- Bovee, C., & Thill, J.V., and Raina, R.L. *Business Communication Today*. New York: Pearson, 2016.
- Lata, Pushp, and Sanjay Kumar. *Communication Skills*. 2nd ed. New Delhi: OUP, 2019.
- Lehman, C. M., Dufrene D. D., and Sinha, M. *BCOM: The South Asian Perspective on Business Communication*. New Delhi: Cengage Learning, 2016.
- Monippally, Matthukutty, M. *Business Communication: From Principles to Practice*. New Delhi: McGraw Hill Pub., 2018.
- Mukerjee, H. S. *Business Communication: Connecting at Workplace*. New-Delhi: Oxford University Press, 2012.
- Murphy, H. A., Hildebrandt, H.W., and Thomas, J.P. *Effective Business Communication*. Boston: McGraw-Hill Companies, 1997.
- Post, Emily. *The Etiquette Advantage in Business*. New York: Collins, 2005.

- Ramesh, Gopalaswamy, and Mahadevan Ramesh. *The Ace of Soft Skills: Attitude, Communication and Etiquette for Success*. Noida: Pearson, 2019.
- Sandra, M. O. *Handbook of Corporate Communication and Strategic Public Relations: Pure and Applied*. New Delhi: Routledge, 2004.
- Sinha, K. K. *Taxmann's Business Communication*. 4th Revised ed. New Delhi: Taxmann's Pub., 2018.
- Taylor, Grant. *English Conversation Practice*. Indian ed. Chennai: McGraw Hill Education Pvt. Ltd., 2017.

Professional Communication Practical

Course Code: 24PENG02

Course Credit: 01 (0-1-0)
Max. Marks: 50 (35I+15E)

Objective: This course is designed to strengthen the communication abilities of the learners by providing them hands-on practice.

Learning Outcomes: After completing this course, the learners will be able to

- 1) Demonstrate knowledge and understanding of a range of professional or public communication situations.
- 2) Perform effectively in diverse professional and public communication situations like interviews and negotiations, drafting emails and resume etc.

Details:

1. Situational Conversations
2. Listening Skills
3. Resume Writing
4. Mock Interviews
5. Group Discussion
6. Presentation Skills
7. Negotiation Skills
8. Email Writing
9. Public Speaking
10. Extempore Speech

Note: The teacher should play the role of the facilitator and allow the learners maximum time to practice these activities. The focus should be primarily on helping the learners overcome the LSWR barrier and gradually move towards honing these skills to enable the learners use them in professional communication

Subject: Mathematical Foundations of Computer Applications (26PMCA01)

Category: Program Core | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Understand fundamental mathematical concepts used in computing.
2. Apply set theory and logic to solve computational problems.
3. Analyze relations, functions, and combinatorics.
4. Use graph theory for modeling and solving problems.
5. Solve recurrence relations and understand their role in algorithms.

Unit-wise Syllabus:

Unit 1: Sets, Relations and Functions

Sets, finite and infinite sets, power set, universal set, set operations (union, intersection, difference, complement), Venn diagrams, laws of set theory, Cartesian product of sets, relations, types of relations (reflexive, symmetric, transitive, antisymmetric, equivalence), matrix and digraph representation of relations, closures of relations, partial ordering, Hasse diagrams, functions, types of functions (one-one, onto, bijective), composition of functions, inverse functions, identity function, binary operations

Unit 2: Propositional and Predicate Logic

Propositions, logical connectives, truth tables, tautology, contradiction, contingency, algebra of propositions, logical equivalence, laws of logic, conditional and bi-conditional statements, rules of inference, arguments, normal forms (CNF and DNF), predicates, quantifiers (universal and existential), free and bound variables, translating statements into predicate logic, rules of inference for quantified statements, direct proof, indirect proof, proof by contradiction, contrapositive

Unit 3: Combinatorics and Recurrence Relations

Addition principle, multiplication principle, permutations, combinations, permutations with repetition, combinations with repetition, binomial theorem, Pascal's triangle, pigeonhole principle, generalized pigeonhole principle, recurrence relations, linear recurrence relations with constant coefficients, homogeneous recurrence relations, non-homogeneous recurrence relations, generating functions

Unit 4: Graph Theory – Trees, Connectivity, Traversal

Graphs, types of graphs (simple, multigraph, directed, weighted), degree of vertex, path, cycle, connected graphs, disconnected graphs, adjacency matrix, incidence matrix, cut vertex, bridge, Eulerian path and circuit, Hamiltonian path and circuit, trees, rooted trees, binary trees, tree traversal (inorder, preorder, postorder), spanning trees, minimum spanning trees, Kruskal's algorithm, Prim's algorithm

Unit 5: Algebraic Structures – Groups, Rings, Monoids, Lattices

Binary operations, closure, associativity, identity, invertibility, semigroups, monoids, submonoids, group definition, subgroups, cyclic groups, cosets, Lagrange's theorem, group homomorphism, group isomorphism, rings, commutative rings, rings with unity, subrings, ring homomorphism, integral

domains, fields, posets, lattices, bounded lattices, complemented lattices, distributive lattices, modular lattices

Textbooks:

1. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", McGraw Hill
2. Tremblay and Manohar, "Discrete Mathematical Structures", McGraw Hill

Reference Books:

1. C.L. Liu, "Elements of Discrete Mathematics"
2. S. Lipschutz, "Discrete Mathematics", Schaum Series
3. J.P. Tremblay, "Mathematical Structure for Computer Science"

Subject: Data Mining and Applications (26PMCA02)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Understand basic concepts of data mining and knowledge discovery.
2. Apply association rule mining, clustering and classification techniques.
3. Evaluate the performance of data mining models.
4. Use data pre-processing techniques for cleaning and transformation.
5. Implement basic mining algorithms on real-world datasets.

Unit-wise Syllabus:

Unit 1: Introduction to Data Mining

Data mining definition, knowledge discovery in databases (KDD) process, steps in KDD (selection, preprocessing, transformation, data mining, interpretation/evaluation), difference between KDD and data mining, types of data mining tasks (descriptive and predictive), data mining functionalities, challenges in data mining, major issues in data mining, applications of data mining in business, healthcare, finance, cybersecurity, and social media

Unit 2: Data Preprocessing

Data quality issues, need for data preprocessing, data cleaning techniques (handling missing values, noisy data, outliers), data integration (schema integration, entity identification, redundancy), data transformation (normalization, smoothing, aggregation, generalization), data reduction (dimensionality reduction, data cube aggregation, numerosity reduction), feature selection, discretization and concept hierarchy generation

Unit 3: Classification Techniques

Introduction to classification, supervised learning, decision tree induction, ID3 algorithm, attribute selection measures (information gain, gain ratio, Gini index), pruning techniques, rule-based classification, naïve Bayes classifier, Bayes theorem, advantages and limitations of decision trees and naïve Bayes, model evaluation metrics (accuracy, precision, recall, F1-score, confusion matrix), cross-validation

Unit 4: Clustering

Introduction to clustering, types of clustering (partitional, hierarchical, density-based), K-Means clustering, initialization and convergence, hierarchical clustering (agglomerative and divisive), distance measures (Euclidean, Manhattan, cosine), dendrograms, DBSCAN algorithm, density-reachability and density-connectivity, comparison of clustering algorithms, applications of clustering in pattern recognition, image analysis, customer segmentation

Unit 5: Association Rule Mining

Introduction to association rules, frequent itemsets, support and confidence, Apriori algorithm, candidate generation and pruning, limitations of Apriori, FP-Growth algorithm, pattern growth approach, comparison of Apriori and FP-Growth, interestingness measures (lift, conviction, leverage), evaluation of association rules, applications of association rule mining in market basket analysis, recommendation systems

Textbooks:

1. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques"
2. Pang-Ning Tan, "Introduction to Data Mining"

Reference Books:

1. Arun K. Pujari, "Data Mining Techniques"
2. Margaret Dunham, "Data Mining: Introductory and Advanced Topics"
3. Ian H. Witten, "Data Mining: Practical Machine Learning Tools and Techniques"

Subject: Data Mining and Applications Lab (26PMCA03)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Pre-process data for mining tasks.
2. Implement classification algorithms like ID3 and Naive Bayes.
3. Perform clustering using K-Means and DBSCAN.
4. Discover frequent patterns and generate association rules.
5. Use tools like WEKA/Python for implementing mining techniques.

List of Experiments:

- 1 Implement a program to demonstrate the KDD process using a sample dataset
- 2 Case study/report on data mining applications in domains like healthcare, retail, or social media
- 3 Handle missing values in a dataset using techniques like mean/mode imputation and deletion
- 4 Perform data cleaning and noise removal using outlier detection techniques
- 5 Implement data normalization techniques (Min-Max, Z-score)
- 6 Demonstrate dimensionality reduction using Principal Component Analysis (PCA)
- 7 Implement a Decision Tree classifier and visualize the tree structure
- 8 Build and evaluate a Naïve Bayes classifier using a real-world dataset
- 9 Compare performance of Decision Tree and Naïve Bayes using metrics and cross-validation
- 10 Implement K-Means clustering and visualize the clusters

- 11 Perform Hierarchical Clustering and generate a dendrogram
- 12 Apply DBSCAN clustering and interpret clusters based on density
- 13 Apply Apriori algorithm to a transactional dataset and generate frequent itemsets and rules
- 14 Implement FP-Growth algorithm and compare with Apriori in terms of performance
- 15 Evaluate association rules using support, confidence, lift, and conviction

Textbooks:

3. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques"
4. Pang-Ning Tan, "Introduction to Data Mining"

Reference Books:

4. Arun K. Pujari, "Data Mining Techniques"
5. Margaret Dunham, "Data Mining: Introductory and Advanced Topics"
6. Ian H. Witten, "Data Mining: Practical Machine Learning Tools and Techniques"

Subject: Programming with C (26PMCA04)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Understand structure and syntax of C programming language.
2. Develop programs using control structures and functions.
3. Apply arrays, strings, pointers and structures in solving problems.
4. Implement file handling in C.
5. Debug and test C programs effectively.

Unit-wise Syllabus:

Unit 1: Introduction to C

History and structure of C program, variables, constants, keywords, data types, type conversion and type casting, arithmetic operators, relational operators, logical operators, bitwise operators, increment and decrement operators, assignment operators, conditional operator, input and output functions (`scanf`, `printf`), basic program structure, compilation and execution.

Unit 2: Control Statements and Functions

Conditional statements (`if`, `if-else`, `nested if`, `switch`), loop control structures (`for`, `while`, `do-while`), `break`, `continue`, `goto`, function definition and declaration, function calling and parameter passing (call by value, call by reference), return values, recursion, scope and lifetime of variables, storage classes

Unit 3: Arrays, Strings, and Pointers

One-dimensional and two-dimensional arrays, array initialization and memory layout, strings and string handling functions (`strlen`, `strcpy`, `strcmp`, `strcat`), character arrays vs string literals, introduction to pointers, pointer arithmetic, pointers and arrays, pointers to strings, pointers to functions, pointers to arrays

Unit 4: Structures, Unions, and Dynamic Memory Allocation

Structure definition and declaration, accessing structure members, nested structures, array of structures, pointer to structure, structure as function argument, union definition and usage, differences between structures and unions, dynamic memory allocation functions (`malloc`, `calloc`, `realloc`, `free`), memory leaks and best practices

Unit 5: File Handling

Introduction to file handling, file streams, file opening and closing, reading from and writing to text files, file modes (`r`, `w`, `a`, `rt`, `wt`), formatted and unformatted I/O (`fscanf`, `fprintf`, `fgetc`, `fputc`, `fgets`, `fputs`), file error handling (`feof`, `ferror`), command-line arguments, practical examples using file handling

Textbooks:

1. E. Balagurusamy, "Programming in ANSI C"
2. Yashavant Kanetkar, "Let Us C"

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, "The C Programming Language"
2. Byron Gottfried, "Programming with C", Schaum's Outline Series
3. Herbert Schildt, "C: The Complete Reference"

Subject: Programming with C Lab (26PMCA05)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Develop basic programs using input/output and operators.
2. Implement conditional and looping constructs.
3. Work with arrays, strings, and pointers in C.
4. Use structures and unions in programs.
5. Implement file handling operations.

List of practical:

- 1 Write a program to demonstrate the use of variables, data types, and arithmetic operators
- 2 Implement a program to take user input and display output using `scanf` and `printf`
- 3 Write a program using conditional statements (`if`, `if-else`, `switch`)
- 4 Write programs using different loops (`for`, `while`, `do-while`)
- 5 Implement a program to demonstrate the use of `break`, `continue`, and `goto`
- 6 Write a program using functions with parameters and return values
- 7 Implement recursion with examples like factorial, Fibonacci series

- 8 Write a program to declare and initialize arrays, and perform operations on them
- 9 Write a program to perform string operations using standard library functions (strlen, strcpy, etc.)
- 10 Implement a program demonstrating basic pointer operations and pointer arithmetic
- 11 Write a program to show the relationship between arrays and pointers
- 12 Write a program to define a structure and access its members
- 13 Write a program to implement dynamic memory allocation using malloc and free
- 14 Write a program to read from and write to a text file using fopen, fprintf, fscanf, etc.
- 15 Implement a program to accept and use command-line arguments

Subject: Fundamentals of Data Science (26PMCA06)

Category: PC | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Understand the data science process and lifecycle.
2. Perform data wrangling and pre-processing.
3. Apply statistical techniques for data analysis.
4. Visualize and interpret data using basic tools.
5. Understand basics of machine learning techniques.

Unit-wise Syllabus:

Unit 1: Introduction to Data Science

Definition of Data Science, data types (structured, unstructured, semi-structured), levels of data measurement (nominal, ordinal, interval, ratio), components of data science, data science workflow (data collection, cleaning, exploration, modeling, interpretation), role of a data scientist, tools for data science (Python, R, Jupyter, etc.)

Unit 2: Data Wrangling

Handling missing data (deletion, imputation), data cleaning techniques, detecting and handling duplicates, data transformation (log transform, binning, encoding categorical variables), feature scaling techniques (normalization, standardization, min-max scaling, robust scaling), data type conversions, parsing dates and times, data integration

Unit 3: Exploratory Data Analysis (EDA)

Descriptive statistics (mean, median, mode, variance, standard deviation), data distributions (normal, skewed), data summarization using Pandas, frequency tables, cross-tabulation, correlation analysis, outlier detection, handling outliers, detecting patterns and anomalies, use of EDA in hypothesis generation

Unit 4: Data Visualization

Importance of data visualization, plotting with Matplotlib (line plot, bar chart, histogram, pie chart), Seaborn plots (heatmap, boxplot, scatter plot, violin plot), customizing plots (labels, titles, legends, themes), combining multiple plots, interactive plots with Plotly or Dash (optional), creating basic data dashboards

Unit 5: Introduction to Machine Learning

Definition of machine learning, difference between AI, ML, and data science, types of machine learning (supervised, unsupervised, reinforcement), overview of supervised learning (classification and regression), unsupervised learning (clustering), training and testing datasets, overfitting and underfitting, introduction to scikit-learn, simple ML workflow

Textbooks:

1. Joel Grus, "Data Science from Scratch"
2. Cathy O'Neil and Rachel Schutt, "Doing Data Science"

Reference Books:

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn"
2. Wes McKinney, "Python for Data Analysis"
3. Roger D. Peng, "Exploratory Data Analysis with R"

Course Title- Yoga and Health Skills-II

Course Credit:02(L02-T0-P0)
:24UYHS01

Course code

Max. Marks: 100(30I+70E)

Objective: Students will get information about the origin, history and development of Yoga along with different streams of yoga, literature and meditation.

Learning Outcome: After the completion of the course, the learners will be able to

1. To make aware of the definition, history and nature of yoga.
2. Telling information about different school of yoga.
3. Giving information about cleansing technique and yogic diet.
4. Explaining the importance of health.
5. Yogic management of various diseases and the utility of meditation.

Unit	Statement
1	Introduction to Yoga: Meaning and Definition of Yoga, Aim and Objectives of Yoga, Misconceptions of Yoga; Brief knowledge about Streams of Yoga-Ashtang and HathaYoga. Yogic Prayer Mantra. Importance of Yoga in modern era.
2	Yoga practices: Raja Yoga (AshtangaYoga), GyanYoga, Bhakti Yoga, Karma Yoga, Hatha Yoga.
3	Introduction to cleansing technique: Meaning, Definition, Objectives and Classification. Yogic diet: Diet, Yogic Diet, Anti-diet, Balanced diet.
4	Health: Meaning, Definition, aim and objectives, Dincharya (Daily regimen): Meaning, definition and sequential elements, Application of Dincharya, Ritucharya (Seasonal Regimen): Meaning, Definition, Types with their salient features, Season wise Does and Don'ts.

5	Yogic management in health problems: cervical, back pain, diabetes and stress. Meditation: Meaning, types, importance, general instructions and suggestions for meditation, physical, mental and spiritual effects of meditation.
---	--

Text Books

1. Yoga & yogic chikitsa - Singh Prof. Ramharsh ,Chaukhamba Sanskrit pratishthan, Edition 2011
2. Swami Vivekananda: Jnana Yoga, Bhakti Yoga, Karma Yoga, Raja Yoga, Advaita Ashrama, Calcutta, 2002.
3. Prof. Ramharsh Singh -SwashthavrittaVigyan, Chaukhambha Sanskrit Prakashan, Varanasi, 1998.
4. Sriram Sharma Acharya- JivemSharadahShatam, AkhandJyoti Mathura 1998.
5. Prof. Ramharsh Singh-Yogewam Yogic Chitksha, Chaukhambha Sanskrit Prakashan, Varanasi, 1998.
6. SwasthaVrittaVigyanewam Yogic Chiktsha- Dr. RakeshGiri, SikhshaBharti, Uttrakhand.

References Books

1. Swami Kuvalyananda : Asana, Kaivalyadhama, Lonavla, 1993
2. Swami Satyananda Saraswati: Asana, Pranayama, Bandha, Mudra, Bihar School of Yoga, Munger, 2006
3. Basavaraddi, I.V. & others: YOGASANA: A Comprehensive description about Yogasana, MDNIY, New Delhi, 2011.
4. Basavaraddi, I.V. & others: Yogic Sukshma Evam Sthula Vyayama, MDNIY, New Delhi, 2011.

MCA Semester II Detailed Syllabus

Subject: Advanced Data Structures (24PMCA07)

Category: PC | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes

After successful completion of the course, students will be able to:

1. Understand and apply advanced data structures in problem-solving.
2. Analyze the efficiency of different data structures.
3. Implement trees, heaps, and hash tables.
4. Solve problems using graphs and related algorithms.
5. Use appropriate data structures for real-world applications.

Unit-wise Syllabus:

Unit 1: Introduction to Data Structures

Abstract Data Types, complexity analysis, arrays, linked lists – singly, doubly, circular, implementation and applications.

Unit 2: Stacks and Queues

Stack operations and applications – expression conversion, recursion; queue operations – linear, circular, priority queues.

Unit 3: Trees

Binary trees, binary search trees, tree traversals (in-order, pre-order, post-order), AVL trees, threaded binary trees.

Unit 4: Heaps and Hashing

Heap structure, heap sort, priority queue operations, hash tables, hashing techniques – linear probing, chaining, rehashing.

Unit 5: Graphs

Graph representation, traversal algorithms (BFS, DFS), shortest path algorithms (Dijkstra's, Bellman-Ford), minimum spanning tree (Kruskal's, Prim's).

Textbooks

1. Ellis Horowitz & Sartaj Sahni, *Fundamentals of Data Structures in C*, Galgotia.
2. Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk.

Recommended Books

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson.
2. Seymour Lipschutz, *Data Structures (Schaum's Outline)*, McGraw Hill.
3. Yedidyah Langsam et al., *Data Structures Using C and C++*, PHI.

Subject: Advanced Data Structures Lab (24PMCA08)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, the student will be able to:

1. Develop basic programs using input/output and operators.
2. Implement conditional and looping constructs.
3. Work with arrays, strings, and pointers in C.
4. Use structures and unions in programs.
5. Implement file handling operations.

List of Practical:

1. Write a program to implement a Singly Linked List with operations such as insertion, deletion, searching, and display.
2. Implement a Doubly Linked List and perform insertion and deletion at different positions.
3. Write a program to create and manipulate a Circular Linked List with traversal operations.
4. Implement Stack operations using Linked List including push, pop, peek, and display.
5. Write a program to implement Queue operations using Linked List including enqueue, dequeue, and display.
6. Implement a Binary Search Tree (BST) and perform insertion, deletion, searching, and traversal operations.
7. Write a program to perform tree traversals including Inorder, Preorder, and Postorder traversal of a Binary Tree.
8. Implement an AVL Tree and perform insertion with balancing using rotations.
9. Write a program to implement Heap operations including insertion, deletion, and heap construction.
10. Implement Heap Sort for sorting a list of elements.
11. Write a program to implement Hashing using linear probing or chaining techniques.
12. Implement Graph Traversal algorithms using Breadth First Search (BFS).
13. Write a program to perform Graph Traversal using Depth First Search (DFS).
14. Implement Dijkstra's Algorithm to find the shortest path from a source vertex to all other vertices in a graph.
15. Write a program to find the Minimum Spanning Tree (MST) using Prim's Algorithm.

Textbooks

1. Ellis Horowitz & Sartaj Sahni, *Fundamentals of Data Structures in C*, Galgotia.
2. Narasimha Karumanchi, *Data Structures and Algorithms Made Easy*, CareerMonk.

Recommended Books

1. Mark Allen Weiss, *Data Structures and Algorithm Analysis in C*, Pearson.
2. Seymour Lipschutz, *Data Structures (Schaum's Outline)*, McGraw Hill.

3. Yedidyah Langsam et al., *Data Structures Using C and C++*, PHI.

Subject: Object Oriented Programming using Python (26PMCA09)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes

After successful completion of the course, students will be able to:

1. Understand the object-oriented approach in Python.
2. Create and use classes and objects efficiently.
3. Apply encapsulation, inheritance, and polymorphism.
4. Work with files and handle exceptions.
5. Build modular, maintainable Python applications.

Unit-wise Syllabus:

Unit 1: Introduction to Python and OOP Concepts

Python basics, variables, data types, operators, control structures, features of object-oriented programming.

Unit 2: Functions and Modules

Function definition and calling, recursion, default and keyword arguments, modules and packages, importing modules.

Unit 3: Classes and Objects

Creating classes and objects, `__init__` method, instance vs class variables, `self` keyword, access specifiers.

Unit 4: Inheritance and Polymorphism

Types of inheritance, method overriding, `super()` function, abstract classes, duck typing, operator overloading.

Unit 5: File Handling and Exception Management

Opening and reading/writing files, file modes, exception types, `try-except-finally` blocks, raising exceptions, `with` statement.

Textbooks

1. Timothy A. Budd, *Exploring Python*, McGraw Hill.
2. E. Balagurusamy, *Introduction to Computing and Problem Solving with Python*, McGraw Hill.

Recommended Books

1. Mark Lutz, *Learning Python*, O'Reilly.
2. R. Nageswara Rao, *Core Python Programming*, Dreamtech Press.

3. Paul Barry, *Head First Python*, O'Reilly.

Subject: Object Oriented Programming using Python Lab(26PMCA10)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, the student will be able to:

1. Develop Python programs using basic programming constructs and control structures.
2. Apply modular and reusable coding practices using functions and modules.
3. Implement object-oriented programming concepts such as classes, inheritance, and polymorphism in Python.
4. Use file handling and exception management techniques for robust Python applications.
5. Design and build Python-based solutions applying real-world programming principles.

List of Practical Experiments

1. Write a Python program to demonstrate basic I/O operations, use of variables, and different data types.
2. Implement a program using arithmetic, logical, and comparison operators with control flow statements (`if`, `elif`, `else`).
3. Write a Python program using `for` and `while` loops with `break`, `continue`, and `pass` statements.
4. Define and call user-defined functions with default, positional, and keyword arguments.
5. Implement a recursive function to compute factorial or Fibonacci numbers.
6. Create and use Python modules and packages, including importing built-in and user-defined modules.
7. Write a class with an `__init__()` method and demonstrate the use of `self` and instance variables.
8. Differentiate between instance and class variables using an example with multiple objects.
9. Demonstrate access specifiers (`public`, `_protected`, `__private`) through a Python class.
10. Implement single and multiple inheritance, and override a method in the child class.
11. Use the `super()` function to invoke parent class constructors and methods.
12. Create an abstract class using `abc` module and implement its abstract methods in a subclass.
13. Demonstrate duck typing and operator overloading using custom Python classes.
14. Write a program for file handling using `open()`, `read()`, `write()`, and different file modes.
15. Implement exception handling using `try-except-finally`, `raise`, and `with` state

Textbooks

1. Timothy A. Budd, *Exploring Python*, McGraw Hill.
2. E. Balagurusamy, *Introduction to Computing and Problem Solving with Python*, McGraw Hill.

Recommended Books

1. Mark Lutz, *Learning Python*, O'Reilly.
2. R. Nageswara Rao, *Core Python Programming*, Dreamtech Press.

3. Paul Barry, *Head First Python*, O'Reilly.

Subject: Distributed Systems (26PMCA11)

Category: PC | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes

After successful completion of the course, students will be able to:

1. Understand principles and models of distributed computing.
2. Apply communication and synchronization techniques.
3. Evaluate naming, consistency, and replication strategies.
4. Analyze distributed file systems and fault tolerance.
5. Explore modern distributed architectures.

Unit-wise Syllabus:

Unit 1: Fundamentals of Distributed Systems

Definition, goals, types of distributed systems, client-server and peer-to-peer models, transparency, scalability.

Unit 2: Communication in Distributed Systems

Message passing, remote procedure call (RPC), remote method invocation (RMI), group communication, multicast.

Unit 3: Synchronization and Coordination

Logical clocks, Lamport timestamps, vector clocks, mutual exclusion algorithms, election algorithms, distributed transactions.

Unit 4: Naming and File Systems

Name services, distributed file system architecture, file access, file sharing, replication, consistency models.

Unit 5: Fault Tolerance and Emerging Trends

Failure models, reliable communication, recovery, replication strategies, blockchain basics, cloud and edge computing overview.

Textbooks

1. Andrew S. Tanenbaum, *Distributed Systems: Principles and Paradigms*, Pearson.
2. George Coulouris et al., *Distributed Systems: Concepts and Design*, Pearson.

Recommended Books

1. M.L. Liu, *Distributed Computing: Principles and Applications*, Pearson.
2. Pradeep K. Sinha, *Distributed Operating Systems*, PHI.

3. Rajiv Chopra, *Distributed Systems*, S. Chand.

Subject: Operating System(26PMCA12)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes

After successful completion of the course, students will be able to:

1. Understand operating system resource management.
2. Learn CPU scheduling, deadlock handling, page replacement and disk scheduling.
3. Use shell commands efficiently.
4. Learn system calls for process, memory and file management.
5. Analyze paging and segmentation techniques.
6. Compare file naming in Linux and Windows.

Unit-wise Syllabus:

UNIT I: Introduction

Operating system concepts, multitasking, multiprogramming, multiuser systems, multithreading, operating system types, services, architecture, system programs and system calls.

UNIT II: Process Management

Process concept, process scheduling, CPU scheduling algorithms: FCFS, SJF, Priority Scheduling, Round Robin and Multilevel Queue Scheduling.

UNIT III: Memory Management

Logical and physical address space, swapping, contiguous and non-contiguous memory allocation, paging, segmentation, virtual memory, demand paging, page replacement algorithms and demand segmentation.

UNIT IV: Process Synchronization and Deadlocks

Critical section problem, semaphores, deadlock prevention, avoidance, detection and recovery.

File System

File types, access methods, directory structures, allocation methods, disk scheduling, disk management and distributed file systems.

UNIT V: I/O Systems and Operating System Overview

I/O hardware, application I/O interface, kernel, I/O request handling, UNIX system calls, shell interpreter, Windows NT architecture and file system.

Text Books:

1. Operating System Concepts – Silberchatz et al.
2. Modern Operating Systems – Andrew Tanenbaum.
3. Operating Systems Internals and Design Principles – William Stallings.

Reference Books:

1. Operating System by Peterson, 1985, AW.
2. Operating System by Milankovic, 1990, TMH.
3. Operating System Incorporating With Unix & Windows By Colin Ritchie, 1974, TMH.
4. Operating Systems by Mandrik & Donovan, TMH
5. Operating Systems – Advanced Concepts By MukeshSinghal , N.G. Shivaratri, 2003, T.M.H

Subject: Operating System Lab(26PMCA13)**Category: PC | L-T-P: 0-0-2 | Credits: 1****Course Learning Outcomes (CLOs):****At the end of the course, the student will be able to:**

1. Understand how an operating system manages all the resources in a computer system efficiently.
2. Learn algorithms for CPU scheduling, deadlock handling, page replacement and disk scheduling used by an operating system.
3. Use shell commands to efficiently operate a computer system.
4. Learn and use system calls for process creation and termination, inter-process communication, process synchronization, memory management and file system management.

List of Practicals

1. Write a program to implement CPU scheduling for First Come First Serve (FCFS).
2. Write a program to implement CPU scheduling for Shortest Job First (SJF).
3. Write a program to perform Priority Scheduling.
4. Write a program to implement CPU scheduling for Round Robin.
5. Write a program for page replacement policy using (a) LRU (b) FIFO (c) Optimal.
6. Write a program to implement First Fit, Best Fit and Worst Fit algorithm for memory management.
7. Write a program to implement Reader/Writer problem using semaphore.
8. Write a program to implement Banker's Algorithm for deadlock avoidance.

Text Books

1. Operating System Concepts by Silberchatz et al., 5th Edition, 1998, Addison-Wesley.
2. Modern Operating Systems by A. Tanenbaum, 1992, Prentice-Hall.
3. Operating Systems Internals and Design Principles by William Stallings, 4th Edition, 2001, Prentice Hall.

Reference Books

1. Operating System by Peterson, 1985, AW.
2. Operating System by Milankovic, 1990, TMH.
3. Operating System Incorporating with Unix & Windows by Colin Ritchie, 1974, TMH.
4. Operating Systems by Mandrik & Donovan, TMH.
5. Operating Systems – Advanced Concepts by Mukesh Singhal and N.G. Shivaratri, 2003, TMH.

Subject: Cyber Security and Information Assurance (26PMCA14)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes

After successful completion of the course, students will be able to:

1. Identify various types of cyber threats and attacks.
2. Understand network security and secure communication.
3. Apply cryptographic techniques.
4. Implement access control and authentication methods.
5. Analyze legal and ethical aspects of cybersecurity.

Unit-wise Syllabus:

Unit 1: Introduction to Cyber Security

Cyber security fundamentals, cyber crimes, types of attacks – phishing, malware, DoS, advanced persistent threats.

Unit 2: Network Security

Basics of networking, firewalls, IDS and IPS, secure protocols, wireless security, VPNs.

Unit 3: Cryptography

Basic concepts, symmetric and asymmetric encryption, RSA, digital signatures, hash functions, public key infrastructure (PKI).

Unit 4: Access Control and Authentication

Authentication factors, password-based systems, biometrics, access control models (DAC, MAC, RBAC).

Unit 5: Information Assurance and Cyber Laws

Confidentiality, integrity, availability (CIA triad), IT Act 2000, digital forensics basics, security policies, ethics and privacy.

Textbooks

1. William Stallings, *Cryptography and Network Security*, Pearson.
2. Nina Godbole & Sunit Belpure, *Cyber Security: Understanding Cyber Crimes*, Wiley.

Recommended Books

1. Behrouz Forouzan, *Cryptography and Network Security*, McGraw Hill.
2. Charles P. Pfleeger & Shari Lawrence Pfleeger, *Security in Computing*, Pearson.
3. Joseph Migga Kizza, *Computer Network Security and Cyber Ethics*, McFarla

Subject: AGENTIC AI (26PMCA15)
Category: VAC | L-T-P: 1-0-0 | Credits: 1

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Understand the fundamentals of Agentic AI and intelligent agents.
2. Explain the architecture and working of AI agents.
3. Use AI tools and frameworks for agent development.
4. Design simple autonomous agents for task execution.
5. Apply Agentic AI concepts to solve real-world problems.

Unit Wise Syllabus

Unit I: Introduction to Agentic AI: Evolution of Artificial Intelligence, Concept of Agents and Agentic AI, Characteristics of Intelligent Agents ,Types of Agents: Reactive, Deliberative, Hybrid Agents, Applications of Agentic AI

Unit II: Agent Architecture and Environment: Agent Architecture, Agent-Environment Interaction ,Perception, Reasoning, and Action, Rational Agents ,Multi-Agent Systems: Introduction

Unit III: AI Tools and Frameworks: Introduction to Large Language Models (LLMs),AI Assistants and Autonomous Agents, Overview of Agent Development Frameworks, Prompt Engineering Basics, Agent Workflows and Task Automation

Unit IV: Agent Design and Development: Designing Simple AI Agents, Goal-Oriented Agents, Planning and Decision-Making, Memory and Context Handling, Agent Communication and Collaboration

Unit V: Applications and Future Trends: Agentic AI in Education, Healthcare, Business, and Research, AI Automation and Productivity Tools, Ethical Issues in Agentic AI, Responsible AI and Safety, Future Trends and Emerging Technologies

Text Books

1. Russell, S. J., & Norvig, P., **Artificial Intelligence: A Modern Approach**, 4th Edition, Pearson, 2021.
2. Wooldridge, M., **An Introduction to MultiAgent Systems**, 2nd Edition, Wiley, 2009.
3. Marr, B., **Generative AI in Practice: 100+ Amazing Ways Generative Artificial Intelligence is Changing Business and Society**, Wiley, 2024.

Reference Books

1. Wooldridge, M., **Artificial Intelligence: A Very Short Introduction**, Oxford University Press, 2021.
2. Lewis, M., **Prompt Engineering for Generative AI**, O'Reilly Media, 2024.
3. Alammar, J., & Grootendorst, M., **Hands-On Large Language Models**, O'Reilly Media, 2024.

Subject: AGENTIC AI LAB(26PMCA16)
Category: VAC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs)

At the end of the course, students will be able to:

CLO1: Understand the fundamentals of AI agents and agent-based systems.

CLO2: Use Generative AI tools and prompt engineering techniques for solving practical problems.

CLO3: Design and develop simple AI agents for automation and decision-making tasks.

CLO4: Implement chatbot and task-oriented agent applications using available AI platforms.

CLO5: Analyze the effectiveness, limitations, and ethical considerations of Agentic AI solutions.

List of Experiments

1. Generate a professional email using an AI assistant.
2. Create a chatbot for college-related queries.
3. Develop a simple FAQ agent for students.
4. Design an AI timetable generation agent.
5. Create an article/paragraph summarization agent.
6. Build a sentiment analysis agent.
7. Develop an interview-question generation agent.
8. Create a task-planning and to-do list agent.
9. Design a resume-outline generation agent.
10. Develop a recommendation agent for books or movies.

Subject: EMPLOYABILITY SKILLS

Course Credit: 01(1-0-0)

Max. Marks: 50(15I+35E)

Course Code: 24UENG03

Objective: This course will introduce students to Basics of Employability Skills with a focus on Attitude, Communication, Etiquettes etc. to enable them practice or showcase professional behavior in formal context.

Learning Outcomes:

1. Learners will be able to use soft skills effectively.
2. Learners will demonstrate a professional behaviour at workplace.
3. Learners will learn Interview skills with ability to prepare for interviews and perform well in the same.
4. Learners will be able to deal with various types of customers in an effective manner.

Unit Name	• Contents
Soft Skills	• Soft Skills- Introduction to soft skills, Aspects and importance of soft skills. • Personality Development: Types of personality; • SWOT Analysis, Goal Setting

Organizational Behavior	• Types of Behavior, • Emotional Intelligence, • Time Management, • Decision Making • Critical Thinking • Team Intelligence and Leadership
Interview Skills	• Interview- Types, a n d Process, • Resume Writing • Job Application; • Research about Industry and • Mapping of Job Competencies with Personal Skills
Interpersonal Skills	• Workplace communication • Active listening, • Positive Attitude • Negotiation Skills

Suggested Readings:

- Bovee, C., & Thill, J.V., and Raina, R.L. *Business Communication Today*. New York: Pearson, 2016.
- Lata, Pushp, and Sanjay Kumar. *Communication Skills*. 2nd ed. New Delhi: OUP, 2019.
- Lehman, C. M., Dufrene D. D., and Sinha, M. *BCOM: The South Asian Perspective on Business Communication*. New Delhi: Cengage Learning, 2016.
- Monippally, Matthukutty, M. *Business Communication: From Principles to Practice*. NewDelhi: McGraw Hill Pub., 2018.
- Mukerjee, H. S. *Business Communication: Connecting at Workplace*. New-Delhi: Oxford University Press, 2012.
- Murphy, H. A., Hildebrandt, H.W., and Thomas, J.P. *Effective Business Communication*. Boston: McGraw-Hill Companies, 1997.
- Post, Emily. *The Etiquette Advantage in Business*. New York: Collins, 2005.
- Ramesh, Gopalaswamy, and Mahadevan Ramesh. *The Ace of Soft Skills: Attitude, Communication and Etiquette for Success*. Noida: Pearson, 2019.
- Sandra, M. O. *Handbook of Corporate Communication and Strategic Public Relations: Pure and Applied*. New Delhi: Routledge, 2004.
- Sinha, K. K. *Taxmann's Business Communication*. 4th Revised ed. New Delhi: Taxmann's Pub., 2018.
- Taylor, Grant. *English Conversation Practice*. Indian ed. Chennai: McGraw Hill Education Pvt. Ltd., 2017.

Employability Skills Practical

Course Credit: 01(0-1-0)

Course Code: 24UENG04

Max. Marks: 50(35I+15E)

Course Objective: This course is designed to strengthen the communication abilities of the learners by providing them hands-on practice.

Learning Outcomes: After completing this course, the learners will be able to

- 1) Demonstrate knowledge and understanding of a range of professional or public communication situations.
- 2) Perform effectively in diverse professional and public communication situations like interviews and negotiations, drafting emails and resume etc.

Details:

1. Role Play in Business Affairs
2. Group Discussion
3. Resume writing
4. Listening Skills
5. Telephone etiquettes
6. Team building
7. Presentation skills
8. SWOT
9. Goal Setting

Note: The teacher should play the role of the facilitator and allow the learners maximum time to practice these activities. The focus should be primarily on helping the learners overcome the LSWR barrier and gradually move towards honing these skills to enable the learners use them in professional communication situations.

MCA Semester III Detailed Syllabus

Subject: Program Elective 1 Artificial Intelligence(26PMCA18)

Category: PC | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Understand the basic concepts of AI.
2. Demonstrate the use of different searching algorithms and differentiate among their working principles and estimate their time and space complexities.
3. Understand the basics of probabilistic reasoning.
4. Understand and apply logic, knowledge representation and reasoning for real world problems.
5. Implement basic AI programming in Prolog.

Unit wise Syllabus

Unit-1

Introduction: Concept of AI, history, current status, scope, agents, environments, Problem Formulations, Review of tree and graph structures, State space representation, Search graph and Search tree.

Unit-2

Search Algorithms: Random search, Search with closed and open list, Depth first and Breadth first search, Heuristic search, Best first search, A* algorithm, Game Search, Hill climbing, Constraint satisfaction problem, Mean - end analysis, Min - Max Search, Alpha - Beta Pruning, Additional refinements, Iterative Deepening.

Unit-3

Probabilistic Reasoning: Probability, conditional probability, Bayes Rule, Dempster-Shafer theory, Certainty factors, Bayesian Networks- representation, construction and inference, Temporal model, Hidden Markov model.

Unit-4

Logic, Knowledge Representation and Reasoning,: First Order Logic, Propositional logic, Predicate logic, Resolution, Clause form, unification algorithm. Building knowledge-bases, Logic based Reasoning Systems, Semantic Networks, Frames.

Planning: The Planning problem, Planning with state space search, partial order planning, planning graphs, planning with propositional logic, Analysis of planning approaches, Hierarchical planning, conditional planning, Continuous and Multi Agent planning.

Unit -5

Prolog Programming: Introduction to Programming in Logic (PROLOG), Lists, Operators, basic Input and Output.

Text Book(s):

1. Elaine Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill
2. Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach”, 3rd Edition, Prentice Hall

References books:

1. Trivedi, M.C., “A Classical Approach to Artificial Intelligence”, Khanna Publishing House, Delhi
2. Saroj Kaushik, “Artificial Intelligence”, Cengage Learning India, 2011.

Subject: Program Elective I Artificial Intelligence Lab(26PMCA24)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

6. Understand basic syntax, facts, and rules of Prolog.
7. Implement classic AI search techniques and logic puzzles.
8. Master recursion, unification, and list manipulation algorithms.
9. Design simple rule-based expert systems and knowledge bases.
10. Apply advanced predicates and grammar rules to solve constraints.

List of Experiments:

1. Write a prolog program to calculate the sum of two numbers.
2. Write a Prolog program to implement max(X, Y, M) so that Mf is the maximum of two numbers X and Y.
3. Write a program in PROLOG to implement factorial (N, F) where F represents the factorial of a number N.
4. Write a Prolog program to implement member(A, S) to check whether A is a member of S or not.
5. Write a Prolog program to implement multiply(X1, X2, M) where X1 and X2 denotes the numbers to be multiplied and Mi represents the result.
6. Write a Prolog program to implement GCD of two numbers
7. Write a program in PROLOG to implement palindrome (S) which checks whether a list S is a palindrome or not.
8. Write a Prolog program to implement maxlist(S, M1); minList(S, M) so that Mf is the maximum/minimum number in the list.
9. Write a Prolog program to merge two lists.
10. Write a prolog program to implement insert_nth (I, N, S, R) that inserts an item [into Nth position of list S to generate a list R
11. Write a prolog program to implement delete_nth (N, S, R) that removes the Nth position item of list S to generate a list R.

Subject: Program Elective I Soft Computing(26PMCA19)

Category: PC | L-T-P: 3-1-0 | Credits: 4

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Apply the importance of knowledge representation and processing in intelligent system
2. Understand the characteristics and constituents of soft computing for decision making systems
3. Apply the models of artificial neural systems for classification problems
4. Apply the learning rules and its working principle for computer vision and image processing applications
5. Apply the importance of auto and hetero associative memories for distinct cases of neural network systems

Unit Wise Syllabus

UNIT-1: Introduction to Neural Networks

Introduction: Fundamental concept, evolution of neural networks, models of artificial neural networks, important technologies, applications, McCulloch, Pitts Neuron, linear separability, Hebb network; Supervised learning

network: Perception networks, adaptive linear neuron, multiple adaptive linear neurons, back propagation network, radial basis function network.

UNIT-2: Associative Memory and Unsupervised Learning Networks

Associative memory networks: Training algorithms for pattern association, auto associative memory network, hetero associative memory network, bidirectional associative memory, Hopfield networks, iterative auto associative memory network, temporal associative memory network;

Unsupervised learning networks: Kohonen self-organizing feature maps, learning vector quantization, counter propagation networks, adaptive resonance theory network.

UNIT-3: Fuzzy Logic

Fuzzy logic: Introduction to classical/crisp sets and fuzzy sets, classical/crisp relations and fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets.

Membership functions: Fuzzification, methods of membership value assignments, defuzzification, and Lambda cuts for fuzzy sets and fuzzy relations, defuzzification methods.

UNIT-4: Neural Networks and Genetic Algorithms

Machine Learning Using Neural Network, Adaptive Networks, Feedforward Networks, Supervised Learning Neural Networks, Radial Basis Function Networks, Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance Theory Networks, Arithmetical Operations on Interval, Solving Equations, Interval Logistic Regression, Basics of Genetic Algorithms. Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning: Machine Learning Approach to Knowledge Acquisition

UNIT-5: MATLAB / Python Lib

Introduction to Matlab/Python, Arrays and array operations, Functions and Files, Study of neural network toolbox and fuzzy logic toolbox, Simple implementation of Artificial Neural Network and Fuzzy Logic.

Recent Trends in deep learning, various classifiers, neural networks and genetic algorithm, Implementation of recently proposed soft computing techniques

Suggested Text Book(s):

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2003.
2. J.S.R. Jang, C. T. Sun, Eiji Mizutani, Neuro, "Fuzzy and Soft Computing", PHI, Pearson Education, 1st Edition, 2004.
3. S. N. Sivanandan, S. N. Deepa, "Principles of Soft Computing", Wiley India, 2nd Edition, 2007.

Reference Books:

1. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice Hall, 1995.
2. S. Rajasekaran, G. A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI, 1st Edition, 2003.
3. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill, 3rd Edition, 1997.
4. Stamatis V. Kartalopoulos, "Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications", IEEE Press, PHI, New Delhi, 2004.

Subject: Program Elective I Soft Computing Lab(26PMCA25)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Build McCulloch-Pitts and Perceptron models for simple logic gates.
2. Implement Backpropagation and RBF networks to resolve non-linear data problems.
3. Deploy Hopfield and BAM networks for pattern storage and retrieval tasks.
4. Use SOM and LVQ networks to cluster complex data distributions.

List of Experiments

1. Implement a single-layer McCulloch-Pitts neuron to simulate basic logical gates (AND, OR, NOT).
2. Write a program to implement a Single-Layer Perceptron network for linearly separable logical gates (AND/OR).
3. Implement the Hebbian Learning rule to train a network for pattern association.
4. Build and train a Multi-Layer Perceptron (MLP) network using the Backpropagation algorithm in Python/MATLAB to solve the non-linear XOR problem.
5. Implement a Radial Basis Function (RBF) network for function approximation or pattern classification tasks.
6. Implement a Hopfield Network to store and retrieve binary character patterns (Auto-associative memory).
7. Program a Bidirectional Associative Memory (BAM) network to find associations between pairs of patterns (Hetero-associative memory).
8. Implement a Kohonen Self-Organizing Map (SOM) for clustering or visual dimensionality reduction of a multi-dimensional dataset.
9. Implement Learning Vector Quantization (LVQ) for a labeled multi-class classification problem.
10. Write a program to perform classical fuzzy set operations (Union, Intersection, Complement, and Algebraic Product) on fuzzy memberships

Subject: Program Elective II Explainable AI (26PMCA30)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Understand the need for interpretability and transparency in AI systems.
2. Distinguish between black-box, gray-box, and white-box models.
3. Apply model-specific and model-agnostic methods to explain AI decisions.
4. Evaluate explainability techniques using metrics and human-centric analysis.
5. Design interpretable AI systems with applications in healthcare, finance, and autonomous systems.

UNIT WISE SYALLBUS

Unit I: Introduction to Explainable AI Need for explainability in AI; Black-box vs white-box models; Trust, fairness, and accountability in AI; Stakeholders and XAI; Legal and ethical motivations (e.g., GDPR, AI Act); Interpretability vs accuracy trade-off; Overview of XAI tools and frameworks

Unit II: Interpretable Models and Intrinsic XAI White-box models: Decision trees, linear models, rule-based learners; Feature importance in interpretable models; Global vs local interpretability; Transparent model design and regularization; Advantages and limitations of inherently interpretable models

Unit III: Post-hoc Model-Agnostic Explanation Techniques LIME (Local Interpretable Model-agnostic Explanations); SHAP (Shapley Additive ExPlanations); Partial Dependence Plots (PDP); Individual Conditional Expectation (ICE); Anchors; Counterfactual explanations; Visualization and interpretation techniques

Unit IV: Explanation Techniques for Deep Learning Gradient-based methods (Saliency Maps, Grad-CAM, Integrated Gradients); Layer-wise relevance propagation; Attention mechanisms and interpretability; Surrogate models for CNNs and RNNs; Explanation challenges in large language models (LLMs); Visualization tools for neural networks

Unit V: Applications, Evaluation, and Challenges in XAI XAI in healthcare, finance, and autonomous systems; Human-centric evaluation of explanations (fidelity, comprehensibility, usability); Quantitative metrics for explainability; XAI in fairness and bias mitigation; Challenges and future directions in explainable and trustworthy AI.

Textbooks

1. Christoph Molnar, Interpretable Machine Learning, Leanpub (open access)
2. Ankur Taly et al., Practical Explainable AI using LIME and SHAP, Packt Publishing

Reference Books

1. Sameer Singh, Marco Ribeiro, Carlos Guestrin, Why Should I Trust You? (Paper on LIME)
2. Been Kim et al., Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV)
3. Rajiv Shah, Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Springer

Subject: Program Elective II Explainable AI Lab(26PMCA38)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Implement interpretable models and visualize decision logic.
2. Apply model- agnostic techniques like LIME and SHAP for local and global explanations.
3. Interpret complex deep learning models using saliency-based and gradient-based techniques.
4. Evaluate the effectiveness of explanations using both qualitative and quantitative metrics.
5. Develop transparent and accountable AI applications in real-world domains.

List of Experiments

1. Build a decision tree model and visualize the decision paths for interpretability.
2. Train a logistic regression and a random forest model, and analyze feature importance.
3. Use PDP to understand global feature effects on predictions in a black-box model.
4. Visualize ICE plots for local interpretability across different input instances.
5. Use the LIME library to explain individual predictions for tabular data.

6. Generate SHAP values and summary plots for tree-based or neural network models.
7. Apply both techniques on a classifier and compare their explanation behavior.
8. Use Anchor explanations to generate high-precision if-then rules from a classifier.
9. Train a CNN and use saliency maps to visualize important input pixels.
10. Apply Grad-CAM to highlight regions in an image that influenced CNN predictions.
11. Implement Integrated Gradients to attribute importance to input features.
12. Visualize attention weights in a transformer-based model for sequence tasks.
13. Use libraries like DiCE to generate and visualize counterfactual samples.
14. Apply evaluation metrics to judge the quality and usefulness of different explanation methods.
15. Develop an interpretable AI system (e.g., loan approval, medical diagnosis) and explain decisions using multiple XAI techniques.

Subject: Program Elective II Natural Language Processing (26PMCA32)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

Unit Wise Syallbus

Unit I

Introduction: A computational framework for natural language, description of English or an Indian language in the framework, lexicon, algorithms and data structures for implementation of the framework, Finite state automata, the different analysis levels used for NLP (morphological, syntactic, semantic, pragmatic, Recursive and augmented transition networks. Applications like machine translations.

Unit II

Word level and syntactic analysis: Word Level Analysis: Regular Expressions, Finite-State Automata, Morphological Parsing, Spelling Error Detection and correction, Words and Word classes, Part-of Speech Tagging. Syntactic Analysis: Context-free Grammar, Constituency, Parsing-Probabilistic Parsing. Machine-readable dictionaries and lexical databases, RTN, ATN.

Unit III

Semantic Analysis: Meaning Representation, Lexical Semantics, Ambiguity, Word Sense Disambiguation. Discourse Processing: Cohesion, Reference Resolution, Discourse Coherence and Structure. Knowledge Representation, Reasoning.

Unit IV

Natural Language Generation (NLG): Architecture of NLG Systems, Generation Tasks and Representations, Application of NLG. Machine Translation: Problems in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches, Translation involving Indian Languages.

Unit V

Information Retrieval: Design features of Information Retrieval Systems, Classical Non Classical, Alternate Models of Information Retrieval , Valuation Lexical Resources: World Net, Frame Net, Stemmers, POS Tagger.

Suggested Text Books

1. Natural Language understanding by James Allen, Pearson Education, 2002.
2. NLP: A Paninian Perspective by Akshar Bharati, Vineet Chaitanya, and Rajeev Sangal, Prentice Hall, 2016.

Reference Books

1. Meaning and Grammar by G. Chierchia and S. McConnell Ginet, MIT Press, 1990.
2. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition by Daniel Jurafsky and James H. Martin, Pearson Education, 2006.
3. Natural language processing in Prolog by Gazdar, & Mellish, Addison-Wesley.

Subject: Program Elective II Lab Natural Language Processing Lab(26PMCA40)

Category: PC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs):

At the end of the course, student will be able to

1. Preprocess text (tokenization, cleaning) using NLTK/spaCy.
2. Implement lexical pipelines like POS tagging and NER.
3. Create feature representations using N-grams, TF-IDF, and LDA.
4. Build core NLP tools like spell-checkers and text classifiers.
5. Deploy pre-trained language models for translation and summarization

List of Experiments

1. Perform text processing operations: Tokenization, stopword removal, stemming, lemmatization using NLTK/spaCy.
2. Write a program for Text Normalization using case folding and punctuation removal.
3. Perform a comparative study by implementing POS tagging using NLTK and Spacy.
4. Write a program to extract named entities using spaCy and visualize them.
5. Write a program to compute word frequency in the corpus. Plot a bar graph for the top five most occurring words.
6. Apply TF-IDF on the text corpus using Scikit learn.
7. Implement bigram and trigram on the text corpus.
8. Perform topic modeling using LDA technique.
9. Implement a basic spell checker using edit distance.
10. Classify text into emotions like joy, anger, sadness using datasets like Emotion Dataset.
11. Perform Text Generation using N-grams based on Markov Chain.
12. Write a program to translate English to French using pre-trained models.
13. Write a program to summarize text using T5 or BART.
14. Write a program to train a model to detect custom entities (e.g., product names).
15. Develop a simple language detection model using character n-grams

Subject: SOFTWARE TESTING & QUALITY ASSURANCE (26PMCA45)

Category: PC | L-T-P: 3-0-0 | Credits: 3

Course Learning Outcomes (CLOs):

At the end the course, student will be able to

1. Knowledge of various Software Testing techniques.
2. Apply Software Testing Strategies and Metrics for Software testing.
3. Implement Object Oriented Testing strategies.
4. Use of Software Quality Assurance.
5. Implement Quality management standards and methods.

Unit-wise Syllabus:

UNIT I

Introduction to Software Testing and Quality Assurance: Fundamentals of Software Testing ,Need and Importance of Software Testing ,Software Quality Concepts and Quality Assurance ,Verification and Validation ,Errors, Faults, Failures, and Defects ,Software Development Life Cycle (SDLC) and Testing Life Cycle (STLC) ,Principles of Software Testing

UNIT-II

Testing Techniques and Test Design: Black Box Testing Techniques , Equivalence Class Partitioning , Boundary Value Analysis ,Decision Table Testing ,State Transition Testing ,White Box Testing Techniques :Statement Coverage ,Branch Coverage ,Path Coverage ,Grey Box Testing,integration, incremental, System, alpha, Beta testing etc; Comparative evaluation of techniques: Testing tools; Dynamic analysis tools, test data generators, Debuggers, test drivers etc. Technical Metrics for Software: Quality Factors, framework; Metrics for analysis, design, testing source code.

UNIT-III

Object Oriented Testing: Introduction to Object Oriented testing, Path Testing, State Based Testing, Class Testing, Testing Web Applications: Web testing, Functional Testing, User interface Testing, Usability Testing, Configuration and Compatibility Testing, Security Testing, Performance Testing, Database testing, Post Deployment Testing.

UNIT-IV

Software Quality Assurance : Software Quality, Software Quality Challenges, Software Quality factors. Software Quality Assurance: concept, components, importance and essence; FTR, structured walk through technique etc.

UNIT-V

Software Quality Management Standards: Management and its role in Software Quality Assurance, Quality Standards: ISO 9000 and Companion ISO Standards, CMM, CMML, Quality Audits and Reviews Emerging Trends in Software Testing :AI in Testing ,Agile Testing ,DevOps and Continuous Quality Assurance

Text Books

1. Naresh Chauhan “Software Testing: Principles and Practices” Oxford University Press
2. Foundations of Software Testing: ISTQB Certification – Dorothy Graham, Erik Van Veenendaal, Isabel Evans, and Rex Black, Cengage Learning.
3. Software Testing – Ron Patton, Pearson Education / Sams Publishing.

Reference Books

1. Introduction to Software Testing – Paul Ammann and Jeff Offutt, Cambridge University Press.
2. Software Testing and Quality Assurance: Theory and Practice – Kshirasagar Naik and Priyadarshi Tripathy, Wiley India.
3. Software Testing: A Craftsman's Approach – Paul C. Jorgensen, Auerbach Publications.
4. Effective Methods of Software Testing – William E. Perry, Wiley Publishing.

Subject: EXCEL & POWER BI FOR DATA VISUALIZATION (26PMCA46)

Category: VAC | L-T-P: 1-0-0 | Credits: 1

Course Learning Outcomes (CLOs)

At the end of the course, students will be able to:

- 1. Understand the fundamentals of data visualization and business intelligence.**
- 2. Apply Microsoft Excel tools for data organization, analysis, and visualization.**
- 3. Create charts, dashboards, and reports using Excel.**
- 4. Develop interactive visualizations and dashboards using Power BI.**
- 5. Interpret and communicate insights from data for decision-making.**

Unit wise Syllabus

Unit I: Introduction to Data Visualization: Introduction to Data, Information, and Business Intelligence ,Importance of Data Visualization ,Types of Data Visualization ,Overview of Microsoft Excel and Power BI ,Applications of Data Visualization

Unit II: Data Handling using Excel:Data Entry and Formatting ,Sorting and Filtering Data ,Conditional Formatting,Data Validation ,Working with Tables

Unit III: Data Analysis in Excel:Basic Formulas and Functions ,Statistical Functions ,Logical Functions (IF, AND, OR) Pivot Tables and Pivot Charts ,Data Cleaning Techniques

Unit IV: Power BI Fundamentals:Introduction to Power BI Desktop ,Data Import and Transformation ,Power Query Editor , Data Modeling Concepts ,Creating Basic Reports

Unit V: Dashboard Development and Reporting:Data Visualization Techniques in Power BI ,Creating Interactive Dashboards ,Slicers and Filters ,Publishing and Sharing Reports ,Business Applications and Case Studies

Text Books

1. Walkenbach, J., **Microsoft Excel 365 Bible**, Wiley Publications, Latest Edition.
2. Alexander, M. & Kusleika, R., **Excel 2021 Power Programming with VBA**, Wiley Publications.
3. Russo, M. & Ferrari, A., **Introducing Microsoft Power BI**, Microsoft Press, Latest Edition.
4. Jelen, B., **Power BI for the Excel Analyst**, Holy Macro! Books, Latest Edition.

Reference Books

1. Powell, S. G., Baker, K. R., & Lawson, B., **Management Science: The Art of Modeling with Spreadsheets**, Wiley.
2. Winston, W. L., **Microsoft Excel Data Analysis and Business Modeling**, Microsoft Press.
3. Ferrari, A. & Russo, M., **The Definitive Guide to DAX**, Microsoft Press.
4. Collie, R. & Singh, A., **Power Pivot and Power BI**, Holy Macro! Books.
5. Few, S., **Information Dashboard Design**, Analytics Press.

Subject: Excel & Power BI for Data Visualization Lab(26PMCA47)

Category: VAC | L-T-P: 0-0-2 | Credits: 1

Course Learning Outcomes (CLOs)

At the end of the course, students will be able to:

- 1: Apply Excel tools and functions for data organization, analysis, and reporting.**
- 2: Perform data cleaning, transformation, and visualization using Excel and Power BI.**
- 3: Create charts, Pivot Tables, and dashboards for business decision-making.**
- 4: Develop interactive reports and business intelligence solutions using Power BI.**
- 5: Analyze real-world datasets and communicate insights through effective visualizations.**

List of Experiments

1. Create and format a Student Performance Analysis Sheet. Calculate Total, Percentage, Grade, and CGPA using Excel functions.
2. Develop an Employee Payroll Management Sheet to calculate Gross Salary, Deductions, Tax, and Net Salary.
3. Perform Sales Data Analysis using Sorting, Filtering, and Advanced Excel Functions.
4. Design an Attendance Monitoring System using Conditional Formatting and Data Validation.
5. Create and analyze Pivot Tables and Pivot Charts for business datasets.
6. Perform Data Cleaning and Transformation using Excel and Power Query.
7. Create visualizations (Bar Chart, Pie Chart, Line Chart, Histogram) for customer or sales data.
8. Import datasets into Power BI and perform data modeling and relationship creation.
9. Develop an Interactive Placement Analysis Dashboard using Power BI with KPIs, Filters, and Slicers.
10. Develop your department Analytics Dashboard incorporating attendance, results, and placement statistics

SUBJECT	Understanding India	Credit	Hours	I	E	T
CODE	24UIN01	2	30	30	70	100
CATEGORY	Value Added Course					
COURSE OBJECTIVE	This course aims to provide students with a comprehensive understanding of India by examining its geographical, historical, cultural, social, and political dimensions.					
COURSE OUTCOMES	On completion of this course, students will be able to:					
	CO1: Critically analyze India’s geographical position on the world map, including its neighboring countries, and articulate the diverse geographical features of India.					
	CO2: Demonstrate an understanding of the concept of Bharatvarsha and evaluate the key elements of ancient Indian literature and religious philosophies, including Vedic traditions, Buddhism, and Jainism.					
	CO3: Assess the significance of oral narratives such as myths, tales, and folklore, and examine the cultural richness of tribal communities within India.					
	CO4: Students will be able to Analyze the continuity and transformation of India’ over time, with a focus on caste, community, class, and gender dynamics.					
CO5: Students will able to explain the importance of India's freedom struggle, the making of the Indian Constitution, and the key features of the Constitution, such as Fundamental Rights and Duties.						
Course Content						
Unit I	Geography of India					
	- India on the map of the world and its neighboring countries - Geographical diversities					
Unit II	History of India					
	- The idea of Bharatvarsha, Ancient Indian literature- Sanskrit, Pali, Prakrit, Tamil - Religions and philosophies of ancient India: Vedic, Buddhism, Jainism					
Unit III	Communicating Culture					
	- Oral narratives: Myths, tales and folklore - Introduction to the Tribal Cultures of India					
Unit IV	Indian Social Structure					
	Continuity and Change of the Indian Social Structure: Caste, Community, Class, and Gender					
Unit V	Understanding Indian Polity					
	- India’s Nationalism and Freedom Struggle - Making of the Indian Constitution - Basic features of the Indian constitution: Fundamental rights and Duties, Directive Principles					

Reading List

Unit I: Geography of India

1. Tirtha, Ranjit 2002: Geography of India, Rawat Publs., Jaipur & New Delhi.
2. Tiwari, R.C. (2007) Geography of India. Prayag Pustak Bhawan, Allahabad

Unit II: History of India

1. <https://iksindia.org>
2. Husain, S. Abid. (2003). *The National Culture of India*, National Book Trust, New

3. Delhi.
The Cultural Heritage of India Series, 8 Volumes (2002), Ramakrishna Mission Institute, Calcutta.

Unit III: Communicating Culture: Tellings, Representations, and Leisure

1. Kanak Mital, "A Santhal Myth, Five Elements" & M.D. Subash Chandran, "Peasant Perception of Bhutas, Uttara Kannada" in Prakrati, *The Integral Vision*, Vol. 1 (Primal Elements – The Oral Tradition, edited by Baidyanath Saraswati), pp. 119-126; 151-166.
2. K. Ramanujan, "'A Flowering Tree': A Woman's Tale," *Oral Tradition*, 12/1 (1997): 226-243.
3. Stuart H. Blackburn, "The Folk Hero and Class Interests in Tamil Heroic Ballads," *Asian Folklore Studies*, Vol. 37, No. 1 (1978), pp. 131-149.
4. Beatrix Hauser, "From Oral Tradition to 'Folk Art': Reevaluating Bengali Scroll Paintings," in *Asian Folklore Studies*, Vol. 61, No. 1 (2002), pp. 105-122.
5. Komal Kothari, "Myths, Tales and Folklore: Exploring the Substratum of Cinema" pdf.

Unit IV: Indian Social Structure

1. Singh, Y. (1968). *Caste and Class: Some Aspects of Continuity and Change*. *Sociological Bulletin*, 17(2), 165–186. <https://doi.org/10.1177/0038022919680205>
2. Singh, Y. (1986). *Modernization of Indian Tradition: A Systemic Study of Social Change*. India: Rawat Publications.
3. Gupta, D. (2000). *Interrogating caste: understanding hierarchy and difference in Indian society*. India: Penguin Books.
4. Xaxa, V. (2008). *State, Society, and Tribes: Issues in Post-colonial India*. India: Dorling Kindersley (India), licensees of Pearson Education in South Asia.
5. Uberoi, P. (1994). *Family, Kinship, and Marriage in India*. India: Oxford University Press.
6. Robinson, R. (2004). *Sociology of Religion in India*. India: SAGE Publications.
7. Srinivas, M. N. (2000). *Caste: Its 20Th Century Avatar*. India: Penguin Books Limited.
8. Jamil, G. (2021). *Women in Social Change*. SAGE Publishing India.
9. Bhasin, K. (2000). *Understanding Gender*.

Unit V: Understanding Indian Polity

1. Madhav Khosla. *The Indian Constitution*. New Delhi, Oxford University Press, 2012.
2. Ramachandra Guha. *Makers of Modern India*. Cambridge, Mass., The Belknap Press of Harvard University Press, 2013.
3. Venkataraghavan Subha Srinivasan. *The Origin Story of India's States*. Penguin Random House India Private Limited, 26 Oct. 2021.
4. J Sai Deepak. *India That Is Bharat: Coloniality, Civilisation, Constitution*. New Delhi, Bloomsbury, 2021.

