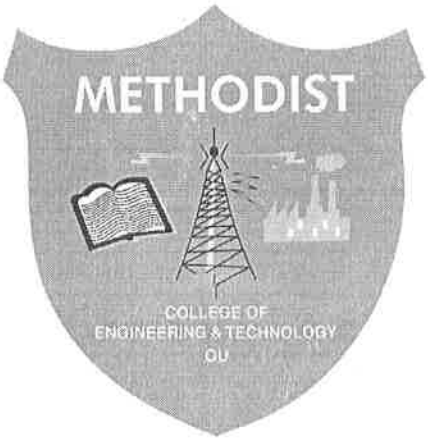


Proposed Program Structure for

B. E

CSE (Artificial Intelligence)

Scheme of Instruction and Examination

Proposed for the Academic Years 2024 - 2028

Scheme of Instruction & Examination
B. E. – CSE (ARTIFICIAL INTELLIGENCE)

S. No	Course Code	Course Title	CSE(AI) Semester - I						Credits
			Scheme of Instruction				Scheme of Examination		
			Hours Per week			Duration in Hrs	Maximum Marks		
			L	T	P/D		CIE	SEE	
Theory Courses									
1	M24BS02HS	Engineering Mathematics –I	3	1	0	4	40	60	4
2	M24BS04HS	Engineering Physics	3	1	0	4	40	60	4
3	M24ES05CS	Programming for Problem Solving	3	0	0	3	40	60	3
4	M24ES02EE	Fundamentals of Electrical and Electronics Engineering	3	0	0	3	40	60	3
5	M24MC03CE	Environmental Science	2	0	0	2	40	60	0
Practical / Laboratory Courses									
6	M24BS52HS	Engineering Physics Lab	0	0	3	3	40	60	1.5
7	M24ES54CS	Programming for Problem Solving Lab	0	0	2	2	40	60	1
8	M24ES52EE	Fundamentals Of Electrical and Electronics Engineering Lab	0	0	2	2	40	60	1
9	M24ES53CE	Engineering Graphics Lab	1	0	4	5	40	60	3
		Total	**15**	**2**	**11**	**28**	**360**	**540**	**20.5**

S. No	Course Code	Course Title	CSE(AI) Semester - II						Credits
			Scheme of Instruction				Scheme of Examination		
			Hours Per week			Duration in Hrs	Maximum Marks		
			L	T	P/D		CIE	SEE	
Theory Courses									
1	M24BS03HS	Engineering Mathematics –II	3	1	0	4	40	60	4
2	M24BS01HS	Chemistry	3	1	0	4	40	60	4
3	M24HS01HS	English	2	0	0	2	40	60	2
4	M24ES01CS	Data Structures	3	0	0	3	40	60	3
Practical / Laboratory Courses									
5	M24BS51HS	Chemistry Lab	0	0	3	3	40	60	1.5
6	M24HS51HS	English Lab	0	0	2	2	40	60	1
7	M24ES51CS	Data Structures lab	0	0	2	2	40	60	1
8	M24ES56ME	Engineering Workshop Practice	0	0	4	4	40	60	2
9	M24MC04HS	Yoga/NSS/Sports	0	0	2	2	50	-	0
		Total	**11**	**2**	**13**	**26**	**370**	**480**	**18.5**

Scheme of Instruction & Examination
B. E. – CSE (ARTIFICIAL INTELLIGENCE)

CSE(AI) Semester - III

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24BS306HS	Probability and Statistics	3	1	0	4	40	60	4
2	M24PC301AI	Discrete Structures	3	0	0	3	40	60	3
3	M24PC302AI	Operating Systems	3	0	0	3	40	60	3
4	M24PC303AI	Database Management Systems	3	0	0	3	40	60	3
5	M24PC304AI	Statistical Analytics and Computing using Python	3	0	0	3	40	60	3
6	M24MC302HS	Essence of Indian Traditional Knowledge	2	0	0	2	40	60	1
Practical / Laboratory Courses									
7	M24PC351AI	Operating Systems Lab	0	0	2	2	40	60	1
8	M24PC352AI	Database Management Systems Lab	0	0	2	2	40	60	1
9	M24PC353AI	Statistical Analytics and Computing using Python Lab	0	0	2	2	40	60	1
10	M24SE351AI	Skill Enhancement Course - IOT	0	0	2	2	40	60	1
		Total Credits				**26**	**400**	**600**	**21**

CSE(AI) Semester - IV

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24ES401EC	Logic Design and Computer Architecture	3	1	0	4	40	60	4
2	M24PC405AI	Software Engineering	3	0	0	3	40	60	3
3	M24PC406AI	Design and Analysis of Algorithms	3	0	0	3	40	60	3
4	M24PC407AI	Data Science	3	0	0	3	40	60	3
5	M24HS402HS	Human Values and Professional Ethics	2	0	0	2	40	60	2
6	M24PC408AI	MOOCS	0	0	0	0	0	0	3
Practical / Laboratory Courses									
7	M24PC454AI	Data Science Lab	0	0	2	2	40	60	1
8	M24PC455AI	Java Programming Lab	0	0	2*2	4	40	60	2
9	M24PW456AI	Mini Project I - App Development – Android/ Flutter/ Flask	0	0	2	2	40	60	1
		Total Credits				**23**	**320**	**480**	**22**

Dept. of Computer Science
Methodist College of Engg. & Tech
King Koti, Hyderabad.

Head of the Department
Department of CSE
Methodist College of Engg. & Tech
Abids, Hyderabad.

P. V. Sudheer
PROFESSOR
Department of Computer Science & Engineering
University College of Engineering (A)
Osmania University,
Hyderabad-500 007.

Scheme of Instruction & Examination
B. E. - CSE(ARTIFICIAL INTELLIGENCE)

CSE(AI) Semester - V									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24PC509AI	Computer Networks	3	1	0	4	40	60	4
2	M24PC510AI	Artificial Intelligence	3	0	0	3	40	60	3
3	M24PC511AI	Machine Learning	3	0	0	3	40	60	3
4	M24PE5(01-05)AI	Professional Elective – I	3	0	0	3	40	60	3
5	M24OE501xx	Open Elective – I	3	0	0	3	40	60	3
Practical / Laboratory Courses									
6	M24PC557AI	Artificial Intelligence Lab	0	0	2	2	40	60	1
7	M24PC558AI	Machine Learning Lab	0	0	2	2	40	60	1
8	M24SE552AI	Skill Enhancement Course-Node JS/ Angular/React JS/Django	0	0	2	2	40	60	1
9	M24SE553HS	Soft Skills Lab - I	0	0	2	2	40	60	1
10	M24SE554HS	Employability Skills-I	0	0	2	2	40	60	1
		Total Credits				**26**	**400**	**600**	**21**

CSE(Artificial Intelligence) Semester – VI									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24ES602EC	Digital Image Processing	3	0	0	3	40	60	3
2	M24PC612AI	Automata Languages and Compiler Design	3	0	0	3	40	60	3
3	M24PC613AI	Deep Learning	3	0	0	3	40	60	3
4	M24PE6(06-10)AI	Professional Elective – II	3	0	0	3	40	60	3
5	M24PE6(11-15)AI	Professional Elective – III	3	0	0	3	40	60	3
6	M24OE602xx	Open Elective – II	3	0	0	3	40	60	3
7	M24HS604HS	Effective Technical Communication(ETCE)	2	0	0	2	40	60	2
Practical / Laboratory Courses									
8	M24ES651EC	Digital Image Processing	0	0	2	2	40	60	1
9	M24PC659AI	Deep Learning Lab	0	0	2	2	40	60	1
10	M24PW660AI	Mini Project II- Real Time/ Societal Research Project	0	0	2	2	40	60	1
		Total Credits				**26**	**400**	**600**	**23**

Professional Elective - I

1	M24PE501AI	Mobile Computing
2	M24PE502AI	Advance Databases
3	M24PE503AI	Software Project Management
4	M24PE504AI	Principles of Programming Languages
5	M24PE505AI	Big Data Analytics

Professional Elective - II

1	M24PE606AI	Digital Forensics
2	M24PE607AI	Distributed Databases
3	M24PE608AI	Agile Methodologies
4	M24PE609AI	Scripting Languages
5	M24PE610AI	Business Analytics

Professional Elective - III

1	M24PE611AI	Cloud Computing
2	M24PE612AI	Soft Computing
3	M24PE613AI	Software Architecture and Design Patterns
4	M24PE614AI	Full Stack Development-I
5	M24PE615AI	Dev Ops

Scheme of Instruction & Examination
B. E. - CSE(ARTIFICIAL INTELLIGENCE)

CSE(AI) Semester - VII									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24HS7xxHS	Managerial Economics and Financial Accounting	3	0	0	3	40	60	3
2	M24PC714AI	Generative AI	3	0	0	3	40	60	3
3	M24PC715AI	Natural Language Processing	3	0	0	3	40	60	3
4	M24PE7(16-20) AI	Professional Elective – IV	3	0	0	3	40	60	3
5	M24OE703xx	Open Elective - III	3	0	0	3	40	60	3
Practical / Laboratory Courses									
7	M24PC761AI	Generative AI Lab	0	0	2	2	40	60	1
8	M24PC762AI	Natural Language Processing Lab	0	0	2	2	40	60	1
9	M24PW763AI	Project Work – I	0	0	4	4	40	60	2
10	M24PW764AI	Summer Internship	-	-	-	-	40	60	1
		Total Credits				**23**	**400**	**600**	**20**

CSE(AI) Semester - VIII									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24PE8(21-25) AI	Professional Elective – V	3	0	0	3	40	60	3
2	M24MC801HS	Constitution Of India	2	0	0	2	40	60	1
Practical / Laboratory Courses									
3	M24PW865AI	Project Work – II			20	20	80	120	10
		Total Credits				**25**	**160**	**240**	**14**

Professional Elective – IV

1	M24PE716AI	Cyber Security
2	M24PE717AI	Nature Inspired Computing
3	M24PE718AI	Software requirements and Estimation
4	M24PE719AI	Full Stack Development-II
5	M24PE720AI	MLOps

Professional Elective - V

1	M24PE821AI	Blockchain Technologies
2	M24PE822AI	Quantum Computing
3	M24PE823AI	Software Testing Methodologies
4	M24PE824AI	Web and Social Media Analytics
5	M24PE825AI	Data Visualization

DEPARTMENT OF CSE(Artificial Intelligence) – CREDIT STRUCTURE

Category	Sem - I	Sem - II	Sem - III	Sem - IV	Sem - V	Sem - VI	Sem - VII	Sem - VIII	Total
HS		3		2		2	3		10
BS	9.5	9.5	4						23
ES	11	6		4		4			25
PC			15	15	12	7	8		57
PE					3	6	3	3	15
OE					3	3	3		9
PW				1		1	3	10	15
SEC			1		3				4
MC	0	0	1					1	2
Total	**21**	**19**	**21**	**22**	**21**	**23**	**20**	**14**	160
Number of Theory Subjects	5	4	6	6	5	7	5	2	
Number of Practical Subjects	4	5	4	3	5	2	5	1	

Course Summary	Inter-Departmental	Departmental	Total
Total Theory Courses	20	19	39
Total Practical Courses	12	18	30

Dept. of Computer Science
Methodist College of Engg. & Tech
King Koti, Hyderabad.

Head of the Department
Department of CSE
Methodist College of Engg. & Tech
Abids, Hyderabad.

PROFESSOR
Department of Computer Science & Engineering
University College of Engineering (A)
Osmania University
Hyderabad

S. No.	Networks/ Security	Theory And Algorithms	Software and Technology	Web Applications	Applications
1	Mobile Computing	Advance Databases	Software Project Management	Principles of Programming Languages	Big Data Analytics
2	Digital Forensics	Distributed Databases	Agile Methodologies	Scripting Languages	Business Analytics
3	Cloud Computing	Soft Computing	Software Architecture and Design Patterns	Full Stack Development-II	Dev Ops
4	Cyber Security	Nature Inspired Computing	Software requirements and Estimation	Full Stack Development-I	ML Ops
5	Blockchain Technologies	Quantum Computing	Software Testing Methodologies	Web and Social Media Analytics	Data Visualization

OPEN ELECTIVES OFFERED BY CSE(ARTIFICIAL INTELLIGENCE) to OTHER DEPARTMENTS

Open Elective - I

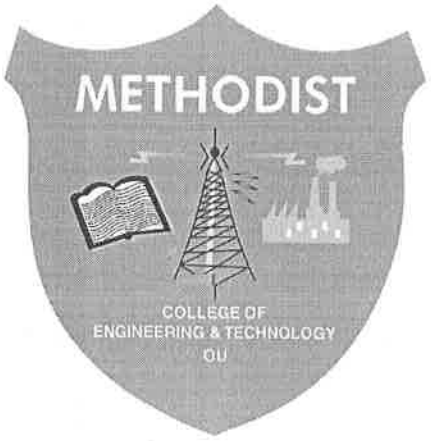
1	M24OEx01AI	Artificial Intelligence

Open Elective - II

1	M24OEx02AI	Machine Learning

Open Elective - III

1	M24OEx03AI	Deep Learning



Proposed Syllabus for

B. E

CSE (Artificial Intelligence)

Proposed for the Academic Years 2024 - 2028

III SEMESTER
SYLLABUS

Scheme of Instruction & Examination
B. E. – CSE (Artificial Intelligence)

CSE(AI) Semester - III									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24BS306HS	Probability and Statistics	3	1	0	4	40	60	4
2	M24PC301AI	Discrete Structures	3	0	0	3	40	60	3
3	M24PC302AI	Operating Systems	3	0	0	3	40	60	3
4	M24PC303AI	Database Management Systems	3	0	0	3	40	60	3
5	M24PC304AI	Statistical Analytics and Computing using Python	3	0	0	3	40	60	3
6	M24MC302HS	Essence of Indian Traditional Knowledge	2	0	0	2	40	60	1
Practical / Laboratory Courses									
7	M24PC351AI	Operating Systems Lab	0	0	2	2	40	60	1
8	M24PC352AI	Database Management Systems Lab	0	0	2	2	40	60	1
9	M24PC353AI	Statistical Analytics and Computing using Python Lab	0	0	2	2	40	60	1
10	M24SE351AI	Skill Enhancement Course - IOT	0	0	2	2	40	60	1
		Total Credits				**26**	**400**	**600**	**21**

Course Code	Course Title						Core / Elective
M24PC301AI	DISCRETE STRUCTURES						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To understand the concepts of Logic, Rules of inference and Quantifiers
2. To explain with examples, the basic terminology of functions, relations, and sets.
3. To understand the basic concept of combinatorics.
4. To relate the ideas of mathematical induction to recursion and recursively defined structures.
5. To develop Graph Algorithms by using the concepts of Graphs and Trees

COURSE OUTCOMES:

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

1. Understand the fundamental concepts of mathematical logic, set theory, algebraic structures, recurrence relations, and graph theory.
2. Apply logical connectives, truth tables, and rules of inference to construct formal proofs and demonstrate set operations, relations, and functions to solve discrete structures problems.
3. Formulate algebraic systems (groups, rings) and combinatorial principles (inclusion-exclusion, pigeonhole) to solve counting problems.
4. Analyze recurrence relations and generating functions to model divide-and-conquer algorithms.
5. Compare graph models (Euler/Hamilton paths) and tree structures (spanning trees) for real-world applications.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2			1				1		2	2		1
CO2	3	3	1		1						2	3	1	1
CO3	3	3	1	1	1						2	3		1
CO4	3	3		2							2	3		1
CO5	3	3	1	3					1		2	3		1

UNIT I

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Quantifiers, universal quantifiers.

Predicates: Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving

UNIT II

Set Theory and Relations: Basic Concepts of Set Theory, Relations and Ordering, Properties of Binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Hasse Diagram.

Functions: Composition of functions, Inverse Functions, Recursive Functions, Lattice and its Properties

UNIT III

Elementary Combinatorics: Basis of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, the principles of Inclusion – Exclusion. Pigeon hole principles and its application.

UNIT IV

Recurrence Relations: Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.

UNIT V

Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Subgraphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Euler Circuits, Planar Graphs, Graph Coloring, Chromatic Numbers

Trees: Introduction to Trees, Properties, Applications of Trees, Tree Traversal, Spanning Trees, Directed Trees, Binary Trees.

TEXTBOOKS

1. Discrete Mathematics and its Applications, Kenneth H. Rosen, V Edition, McGraw Hill Education, 2017.
2. Elements of Discrete Mathematics- A Computer Oriented Approach- C L Liu, D PMohapatra,III Edition, Tata McGrawHill, 2017.
3. Discrete Mathematics for Computer Scientists & Mathematicians, J.L. Mott, A. Kandel,T.P Baker, II Edition, PHI, 2015.

REFERENCE BOOKS

1. Discrete Mathematical Structures Theory and Application- Malik & Sen, I Edition, Cengage Learning, 2012.
2. Discrete Mathematics with Applications, Thomas Koshy, I Edition, Elsevier, 2005.

Course Code	Course Title						Core / Elective
M24PC302AI	OPERATING SYSTEMS						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To provide a comprehensive understanding of the fundamental principles, functions, and services of operating systems in managing hardware and software resources.
2. To enable students to learn various concepts of process management, thread handling, synchronization techniques, and inter-process communication.
3. To impart knowledge on memory management strategies, virtual memory concepts, and page replacement policies to optimize system performance.
4. To develop an understanding of file systems, I/O device management, and storage techniques for efficient data handling and system operations.
5. To expose students to modern operating system features such as virtualization, security mechanisms, and system architectures through real-world case studies like Windows 11.

COURSE OUTCOMES:

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

1. Demonstrate familiarity with fundamental concepts, terminologies, and principles of operating systems and their role in managing system resources.
2. Explain and summarize system structures, process and memory management, file systems, I/O handling, virtualization, and security features as applied in modern systems.
3. Apply suitable algorithms, techniques, and tools to address problems related to process management, memory allocation, file operations, I/O management, and system-level resource utilization.
4. Analyze and differentiate operating system strategies and architectures to assess their efficiency, reliability, and security, referring to practical implementations.
5. Evaluate and justify the selection of various operating system approaches for optimal system performance, resource management, and security, drawing insights from real-world platforms.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2			1				1		2	2		1
CO2	3	3	2	2	2						2	3	1	2
CO3	3	3	3	2	2			1			2	3	2	3
CO4	3	3	2	3	2						2	2	1	2
CO5	3	3		3	2	1	1		2		3	3		2

UNIT I

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

UNIT II

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling Criteria, Scheduling algorithms, multiprocessor scheduling

UNIT III

Process Synchronization: Inter-process Communication, Critical Section, Race Conditions, Mutual Exclusion, Peterson's Solution, Classical problems of synchronization: The Bounded buffer problem, Producer\Consumer Problem, reader's & writer problem, Dinning philosopher's problem. Semaphores, Event Counters, Monitors, Message Passing,

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Methods for Handling: Deadlocks: Deadlock prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT IV

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation, fragmentation and Compaction; Paging: Principle of operation, Page allocation, Hardware support for paging, structure of page table, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory, Hardware and control structures, Locality of reference, Page fault, Working Set, Dirty page/Dirty bit, Demand paging, Page Replacement algorithms, Trashing

UNIT V

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods, Free-space management, directory implementation, efficiency and performance.

Secondary-Storage Structure: Disk structure, Disk scheduling algorithms, Disk Management, RAID structure.

Case Study: "Operating System Case Study – Windows 11: Process and Memory Management, File Systems, I/O Management, Security Features, Virtualization, and System Architecture."

TEXTBOOKS

1. Operating System Concepts Essentials, Abraham Silberschatz, Peter B Galvin, Greg Gagne, IX Edition, Wiley Asia Student Edition, 2017.
2. Operating Systems: Internals and Design Principles, William Stallings, V Edition, Prentice Hall of India, 2016.
3. Modern Operating Systems, Andrew S. Tanenbaum (2007), II edition, Prentice Hall of India, India.

REFERENCE BOOKS

1. Design of the Unix Operating Systems, Maurice Bach, VIII Edition, Prentice-Hall of India, 2009.
2. Understanding the Linux Kernel, Daniel P. Bovet, Marco Cesati, III Edition, O'Reilly and Associates.

UNIT III

Relational Model: Introduction to Relational Model, Basic Structure, Database Schema, Keys, Relational Algebra and Relational Calculus.

Schema Refinement and Normal Forms: Introduction to Schema Refinement, Functional Dependencies, Reasoning about FD

Normal Forms and Normalization: INF,2NF,3NF, BCNF,4NF,5NF, Properties of Decomposition

UNIT IV

Transaction Management: Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability.

UNIT V

Concurrency Control: Lock based Protocols, Timestamp based protocols, Recovery System: Recovery and Atomicity, Log based recovery, Shadow Paging, Recovery with concurrent Transactions, Buffer Management.

Storage and Indexing: File Organizations and Indexing-Overview of Indexes, Types of Indexes, Index Data Structures, Tree structured Indexing, Hash based Indexing.

TEXTBOOKS

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, III Edition, TATA McGraw Hill.
2. Data base System Concepts, Silberschatz, Korth, V Edition, McGraw Hill.
3. Introduction to Database Systems, C.J.Date , VIII Edition, Pearson Education.
4. Database Systems design, Implementation, and Management, Rob & Coronel, IX Edition, Cengage Learning, Inc

REFERENCE BOOKS

1. Fundamentals of Database System, Elmasri Navate, VII Edition, Pearson Education.
2. Database Management System, Mathew Leon, Leo, I Edition, Tata McGraw Hill Education

Course Code	Course Title						Core / Elective
M24PC304AI	STATISTICAL ANALYTICS AND COMPUTING USING PYTHON						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving using C	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To introduce students to the Python programming environment and familiarize them with its syntax, semantics, and core programming constructs.
2. To develop the ability to write Python programs using conditional, looping, and function constructs for solving real-time computational problems.
3. To expose students to different Python data structures such as strings, lists, dictionaries, sets, and tuples with emphasis on their applications.
4. To train students in using external libraries such as NumPy and Pandas for data manipulation, statistical analysis, and basic visualization.
5. To prepare students to build modular, efficient Python programs that include file handling, exception management, and data analysis capabilities

COURSE OUTCOMES:

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

1. Recall and interpret the basic concepts of Python programming, including syntax, variables, data types, and standard libraries.
2. Apply control structures, functions, file operations, and data structures like lists, strings, tuples, sets, and dictionaries to solve basic computational problems.
3. Analyze complex problems using modular and structured programming approaches, handling exceptions and manipulating structured data.
4. Evaluate data using built-in functions, sorting/searching algorithms, and summarize data using NumPy and Pandas libraries.
5. Design and develop Python-based solutions for simple data analysis tasks using appropriate libraries and visualization techniques.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				1						1		2	1
CO2	3	2			2						2		3	2
CO3	3	3	2		2						2		3	3
CO4	2	2	2		3						2		3	3
CO5	2	3	3	2	3						3		3	3

UNIT I

Introduction to Python Programming: Python Interpreter, Parts of Python Language: Identifiers, Keywords, Statements, Variables and Operators, Data Types, Indentation, Comments, Type conversions, type() Function and is Operator, Dynamic and strong Typed language, Command line arguments.

Control Flow Statements: if, if.else, if...elif...else Decision Control Statements, Nested if Statement, the while Loop, the for Loop, the continue and break Statements. Exception Handling.

UNIT II

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List operations, Indexing and slicing in Lists, Built in Function used in List, List Methods, Advanced List processing - list comprehension. Illustrative programs: selection sort, insertion sort, merge sort.

UNIT III

Dictionaries: Creating Dictionaries, Accessing and Modifying, Built-in Functions used in Dictionaries, Dictionary Methods.

Tuples and Sets: Creating tuples, Basic tuple operations, Indexing and slicing, Built-in Functions used in tuples, tuple methods, Sets: Creating sets, Basic set operations, set methods.

Files and Modules: Types of Files, Creating, reading, and writing files, File Methods. Modules and packages: Built-in Modules, creation of user defined modules, Pickle Module, Illustrative programs: word count, copy file, reading and writing CSV files.

UNIT IV

NumPy: Understanding NumPy arrays, NumPy Arrays creation using array() function, Array attributes, Arrays creation with initial placeholder content, Indexing, Basic Arithmetic operations on NumPy Arrays, Mathematical Functions in NumPy, manipulating array shapes, Stacking and Splitting of NumPy Arrays, Broadcasting arrays.

UNIT V

Pandas: Creating pandas Data Frames, understanding pandas Series, Reading, and querying the Quandl data, describing pandas Data Frames, Grouping, and joining pandas Data Frame, working with missing values, creating pivot tables. Overview of Python for Data Science.

TEXTBOOKS

1. Introduction to Python Programming, Gowrishankar S., Veena A, CRC Press, Taylor & Francis Group, 2019.
2. Learning to Program with Python, Richard L. Halterman, Copyright © 2011.
3. A First Course in Probability and Statistics, B. L. S. Prakasa Rao,World Scientific/ Cambridge University Press India, 2009.

REFERENCE BOOKS

1. Introduction to Mathematical Statistics, R. V. Hogg, J. W. McKean and A. Craig, VI Edition, Pearson Education India, 2006.
2. Introduction to statistical learning: With applications to R,Gareth M. James, Springer2013

Course Code	Course Title						Core / Elective
M24PC351AI	OPERATING SYSTEMS LAB						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	40	60	1

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To Learn various system calls in Linux
2. To Learn different types of CPU scheduling algorithms.
3. To Demonstrate the usage of semaphores for solving synchronization problem
4. To Understand memory management techniques and different types of fragmentation.
5. To Learn various disk scheduling algorithms

COURSE OUTCOMES:

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

1. Use different system calls for writing application programs
2. Evaluate the performance of different types of CPU scheduling algorithms.
3. Implement producer-consumer problem, reader-writer's problem, Dining philosopher's problem.
4. Simulate Banker's algorithm for deadlock avoidance.
5. Implement paging replacement and disk scheduling techniques

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2		1	3				2		2	2	1	2
CO2	3	3	2	2	3				1		2	2	3	3
CO3	3	3	3	2	3			1	1		2	2	3	3
CO4	3	3	2	3	3	1					2	2	2	3
CO5	3	3		3	3		1	2	2		3	2	1	2

List of Programs (preferred programming language is C)

Perform a case study by installing and exploring various types of operating systems on a physical or logical (virtual) machine

1. Write C programs to implement UNIX system calls and file management system calls.
2. Write C programs to demonstrate various process related concepts.
3. Write C programs to demonstrate various thread related concepts.
4. Write C programs to simulate CPU scheduling algorithms: FCFS, SJF, Round Robin
5. Write C programs to simulate Intra & Inter-Process Communication (IPC) techniques: Pipes, Messages Queues, Shared Memory.
6. Write C programs to simulate solutions to Classical Process Synchronization Problems: Dining Philosophers, Producer-Consumer, Readers-Writers
7. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
8. Write C programs to simulate Page Replacement Algorithms: FIFO, LRU
9. Write C programs to simulate implementation of Disk Scheduling Algorithms: FCFS, SSTF.
10. Shell programming: creating a script, making a script executable, shell syntax (variables, conditions, control structures, functions, and commands).

Course Code	Course Title						Core / Elective
M24PC352AI	DATABASE MANAGEMENT SYSTEMS LAB						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	40	60	1

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To practice various DDL, DML commands in SQL
2. To write simple and Complex queries in SQL
3. To practice various Functions, Joins &sub queries in SQL
4. To write PL/SQLusing cursors and collections
5. To write PL/SQL using Stored Procedures

COURSE OUTCOMES:

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

1. Design and implement a database schema for a given problem
2. Develop the query statements with the help of structured query language.
3. Populate and query a database using SQL and PL/SQL
4. Develop multi-user database application
5. Design and implement E-R model for the given requirements

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	1	3			2	1	1	2	3	3	3
CO2	3	3	2		3			1			2	3	3	3
CO3	3	3	2	1	3			1			2	3	3	3
CO4	3	3	3	2	3		1	2	1	2	2	3	3	3
CO5	3	3	3	1	3			1			2	3	2	3

List of Programs:

1. Creation of database Tables (exercising the all SQL commands)
2. Simple and complex condition query creation using SQL Plus
3. Creation of database Tables using Integrity constraints and Functions
4. Simple and complex condition query creation using Joins
5. Simple and complex condition query creation using Sub queries and set operators
6. Creation of Views (exercising the all types of views)
7. Writing PL/SQL function and cursors
8. Writing PL/SQL stored procedure and triggers
9. Creation of Forms and reports for student Information, library information, Pay roll etc.
10. Case Study: Design Database for Bank
    => Collect the information Related with Bank organization
    => Draw E-R Diagrams for Bank
    => Reduce E-R Diagrams to tables
    => Normalize your Database up to 3rd Normal form
    => Retrieve Bank information using SQL commands

TEXTBOOKS

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, III Edition, TATA McGraw Hill.
2. Data base System Concepts, Silberschatz, Korth, V Edition, McGraw Hill.

REFERENCE BOOKS

1. Database Management System, Elmasri Navate, Pearson Education.
2. Database Management System, Mathew Leon, Leo

Course Code	Course Title						Core / Elective
M24PC405AI	SOFTWARE ENGINEERING						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To understand the fundamental concepts, principles, and practices of software engineering.
2. To learn various software development process models and project planning techniques.
3. To develop skills in system modelling using UML diagrams and Data Flow Diagrams (DFDs).
4. To gain knowledge of software implementation, testing strategies, and maintenance activities.
5. To familiarize with software quality assurance, configuration management, and recent trends in software engineering.

COURSE OUTCOMES:

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

1. Explain foundational software engineering concepts, including process models, requirement engineering, and UML diagrams.
2. Describe software testing strategies, product metrics, and risk management frameworks.
3. Select appropriate process models and architectural styles for given project scenarios.
4. Compare the effectiveness of validation testing techniques and metrics for software quality assessment.
5. Assess risk management strategies and their impact on software project success.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	1		1		1		1		2	2		2
CO2	3	3	2		2						2	3	2	3
CO3	3	3	2	1	2						2	3	2	3
CO4	3	3	1	2	1						2	3	1	2
CO5	3	3	1	3							2	3		2

UNIT I

Software Engineering Fundamentals: Introduction to Software Engineering: Nature and characteristics of software, Software crisis and software myths, Software Process: Prescriptive process models: Waterfall, Incremental, Spiral, Concurrent, Process assessment and improvement (CMMI overview), Agile Development: Agile manifesto and principles, Agile process models: Scrum, Extreme Programming (XP)

UNIT II

Requirements Engineering and Project Planning: Requirements Engineering: Elicitation, analysis, specification, and validation, Functional and non-functional requirements, Software Requirements Specification (SRS), Software Project Management: Estimation techniques (LOC, Function Points), Scheduling: Gantt charts, PERT, Risk management and mitigation planning

UNIT III

System Modelling Using UML and Data Flow Diagrams: UML Diagrams for System Modelling: Use Case Diagram, Class Diagram, Sequence Diagram, Activity Diagram, State Chart Diagram, Deployment Diagram, Data Flow Diagrams (DFD): DFD symbols and rules, Context-level DFD,

Level 0 and Level 1 DFD, Case-based examples and DFD modelling best practices, Brief Overview of Design Concepts: Abstraction, modularity, cohesion, coupling (only definitions and relevance to modelling)

UNIT IV

Software Implementation, Testing, and Maintenance: Software Construction: Coding practices, standards, documentation, Software Testing: Testing fundamentals and principles, Unit, integration, system, and acceptance testing, White-box and black-box testing, Test case design techniques, Maintenance and Reengineering: Maintenance types, Software evolution and legacy systems, Basics of reverse engineering

UNIT V

Software Quality, SCM, and Case Study: Software Quality Assurance: Product and process quality, Software reviews, audits, and metrics, Software Configuration Management (SCM): Version control, change control, repositories

Emerging Trends: DevOps and CI/CD, Cloud-based development practices, Security in software engineering, Ethics in software development,

Case Study: End-to-end analysis of a software development project, From requirements to design (UML/DFD), testing, and deployment, Students interpret or develop partial artifacts for the given system

TEXTBOOKS

1. Software Engineering: A Practitioner's Approach, Roger S. Pressman, VII Edition, McGraw Hill, 2009
2. Software Engineering. Ian Sommerville, VII edition, Addison-Wesley, 2004.
3. Fundamentals of Software Engineering Rajib Mall, V Edition, PHI, 2009.

REFERENCE BOOKS

1. Software Engineering Fundamentals, Ali Behforooz and Frederick J. Hudson, Oxford University Press, 1996
2. An Integrated Approach to Software Engineering, Pankaj Jalote, III Edition, Narosa Publishing House, 2000
3. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, VIII Edition, John Wiley.

Course Code	Course Title						Core / Elective
M24PC406AI	DESIGN AND ANALYSIS OF ALGORITHMS						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To provide an understanding of AI fundamentals, including problem-solving, search algorithms and game-playing techniques.
2. To teach knowledge representation methods and uncertainty handling techniques.
3. To introduce planning techniques such as hierarchical task network, constraint satisfaction and Markov decision process.
4. To provide an understanding of machine learning techniques for solving real-world problems.
5. To explore AI applications in NLP, computer vision while addressing ethical concerns.

COURSE OUTCOMES:

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

1. Explain fundamental algorithm analysis techniques, design paradigms (divide-and-conquer, greedy, dynamic programming), and complexity classes (P, NP, NP-complete).
2. Apply divide-and-conquer techniques to solve sorting and search problems with recurrence analysis.
3. Implement greedy, dynamic programming, backtracking, and branch-and-bound strategies for optimization and constraint satisfaction problems.
4. Analyze NP-completeness proofs and reductions for classical problems like clique and vertex cover.
5. Evaluate the suitability of different algorithm design paradigms for given problem characteristics.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2			1				1		2	2		1
CO2	3	3	1		1						2	3	1	1
CO3	3	3	2	1	1						2	3	1	2
CO4	3	3		3							2	3		1
CO5	3	3	1	3					1		2	3		1

UNIT I

Introduction & Elementary Data Structures: Introduction, Fundamentals of algorithm (Line Count, Operation Count), Analysis of algorithms (Best, Average, Worst case), Asymptotic Notations (O, Ω, Θ) Recursive Algorithms, Analysis using Recurrence Relations, Master's Theorem.

Review of elementary data structures–Graphs: BFS, DFS, Articulation points, Bi-Connected Components.

Sets: representation, UNION, FIND operations.

UNIT II

Divide-and-Conquer Method: The general method, Binary search, Finding maximum and minimum, Merge sort, Quick sort.

Brute Force: Knapsack, Travelling salesman problem, Convex-Hull.

UNIT III

Greedy Method: Knapsack problem, Minimum spanning trees, Single source shortest path, Job sequencing with deadlines, Optimal storage on tapes, Optimal merge pattern.

Dynamic programming method: All pairs shortest paths, Optimal binary search tress, 0/1 Knapsack problem, Reliability design, Travelling salesman problem.

UNIT IV

Back tracking: N-queen's problem, Graph coloring, Hamiltonian cycles.

Branch-and-bound: FIFO & LC branch and Bound methods, 0/1 Knapsack problem, Travelling salesperson.

UNIT V

NP-Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem, Proofs for NP Complete Problems: Clique, Vertex Cover.

TEXTBOOKS

1. Fundamentals of Computer Algorithms, Horowitz E, Sahni S, II Edition, Universities Press, 2007.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, III Edition, PHI Learning Private Limited, 2012.

REFERENCE BOOKS

1. Algorithm Design: Foundations, Analysis and Internet Examples, Michael T. Goodrich, Roberto Tamassia, I Edition, John Wiley & Sons,2002
2. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, II Edition, Pearson education.

Course Code	Course Title						Core / Elective
M24PC407AI	DATA SCIENCE						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Statistical analysis using Python	3	-	-	-	40	60	3

COURSE OBJECTIVES:

The objective of this course is to make the student

1. Introduce Python and essential libraries used in data science.
2. Enable students to perform data cleaning, transformation, and visualization.
3. Teach core statistical techniques like descriptive and inferential statistics.
4. Develop understanding of machine learning models for classification and clustering.
5. Expose students to applications of data science in network and graph analysis.

COURSE OUTCOMES:

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

1. Understand the fundamental concepts, tools, and processes in data science using Python.
2. Apply Python libraries to manipulate, visualize, and analyze data.
3. Analyze statistical data, models, and clustering to interpret data patterns.
4. Evaluate data hypotheses and models using statistical inference and metrics.
5. Create data science applications using Python for real-world problem-solving.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3										1	1	2	
CO2	3	2									1	2	3	2
CO3	3	3	2								2	3	3	3
CO4	3	3	2	2							2	3	3	3
CO5	3	3	2	2							3	3	3	3

UNIT I

Introduction, Toolboxes: Python, fundamental libraries for data Scientists. Integrated development environment (IDE). Data operations: Reading, selecting, filtering, manipulating, sorting, grouping, rearranging, ranking, and plotting.

UNIT II

Descriptive statistics, data preparation. Exploratory Data Analysis data summarization, data distribution, measuring asymmetry. Sample and estimated mean, variance and standard score. Statistical Inference frequency approach, variability of estimates, hypothesis testing using confidence intervals, using pvalues

UNIT III

Supervised Learning: First step, learning curves, training-validation and test. Learning models generalities, support vector machines, random forest. Examples

UNIT IV

Regression analysis, Regression: linear regression simple linear regression, multiple & Polynomial regression, Sparse model. Unsupervised learning, clustering, similarity and distances, quality measures of clustering, case study.

UNIT V

Network Analysis, Graphs, Social Networks, centrality, drawing centrality of Graphs, PageRank, Ego-Networks, community Detection

TEXTBOOKS

1. Introduction to Data Science a Python approach to concepts, Techniques and Applications, Igual, L;Seghi, S. Springer, ISBN:978-3-319-50016-4 2.
2. Data Analysis with Python A Modern Approach, David Taieb, Packt Publishing, ISBN-9781789950069

REFERENCE BOOKS

1. Python Data Analysis, Armando Fandango, II Edition, Packt Publishing, ISBN:9781787127487

Course Code	Course Title						Core / Elective
M24PC454AI	DATA SCIENCE LAB						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Statistical analysis using Python	-	-	-	2	40	60	1

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To familiarize students with Python programming and IDEs for data handling and manipulation.
2. To implement basic algorithms and data visualization techniques using Python.
3. To develop programs for statistical analysis and exploratory data analytics.
4. To train students in applying classification and clustering methods on datasets.
5. To provide hands-on experience in graph analytics and network-based data science applications.

COURSE OUTCOMES:

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

1. Understand and use Python interactive commands, file handling, and data operations.
2. Apply Python to implement searching, sorting, and data visualization techniques.
3. Analyze datasets using statistical tools such as standard deviation, variance, and data distributions
4. Develop programs for machine learning models like SVM and k-means for data classification and clustering.
5. Evaluate social networks using graph analytics such as community detection, centrality, and PageRank.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				2								2	
CO2	3	2			3								3	
CO3	3	3			3								3	
CO4	3	3	3		3							3	3	
CO5	3	3	3		3							3	3	

LIST OF PROGRAMS:

1. Interactive commands in Python, data operations, file read/write and manipulations
2. Familiarization with IDE in Python
3. Writing programs for standard algorithms of sorting and searching in Python
4. Plotting the data using X-Y graph, Bar chart, and other plotting techniques
5. Perform exploratory data analysis: variance, std. deviation, summarization, distribution, inference
6. Plotting the various distributions for given data sets
7. Classifying and presentation of data using Support Vector Machine
8. Write programs for K-means clustering and presentation for given data sets
9. Write programs on graphs of social networks for community detection
10. Write programs for analysis of graphs to find centrality and PageRank

Course Code	Course Title						Core / Elective
M24PC455AI	JAVA PROGRAMMING LAB						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Programming and Problem Solving	-	-	-	2*2	40	60	2

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To implement various java concepts.
2. To write java programs to solve mathematics, science and engineering problems.
3. To identify compile time and runtime errors, syntax and logical errors
4. To import the essentials of java class library and user defined packages.
5. To develop skills in internet programming using applets and swings

COURSE OUTCOMES:

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

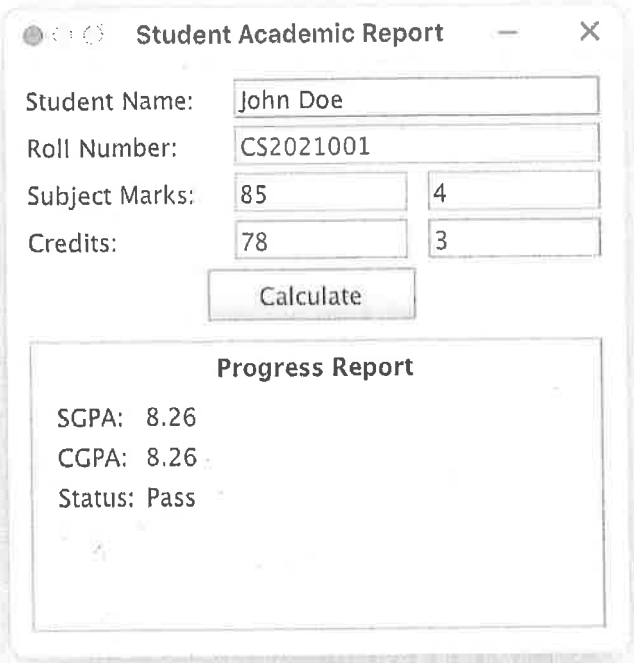
1. To understand the use of OOPs concepts.
2. Develop Java program using packages, inheritance and interface.
3. Develop java programs to implement error handling techniques using exception handling.
4. Develop graphical user interface using AWT.
5. Demonstrate event handling mechanism

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2		3			1	1		2	2	3	2
CO2	3	3	3		3			2			2	3	3	3
CO3	3	3	3	1	3			1			2	3	3	3
CO4	3	3	3		3			2	1		2	3	3	3
CO5	3	3	3		3			2	1		2	3	3	3

List of Programs

1. Implement the concept of classes and objects.
2. Implement Arrays to a given application.
3. Use String and String Tokenizer classes and develop a java programs.
4. Develop a java programs Using interfaces and packages.
5. Develop Java Programs using inheritance.
6. Develop Java programs using Method overloading and method overriding.
7. Develop java programs using Exception handling (using try, catch, throw, throws and finally).
8. Develop java programs using Multithreading (using Thread class and Runnable interface, synchronization).
9. Develop java programs using collections (using list, set, Map and generics).
10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired. [Use Adapter classes]
11. Write a program to perform CRUD operations on the student table in a database using JDBC.
12. CASE STUDY: Design and develop a Java application to calculate and display the SGPA and CGPA of a student. The application should provide a graphical user interface (GUI) using Java

Swings for inputting student details and marks. The calculated SGPA and CGPA, along with the student's progress report, should be displayed using an Applet interface. Additionally, the application should establish a connection with a backend database using JDBC to store and retrieve student academic records.



Course Code	Course Title						Core / Elective
M24PW456AI	MINI PROJECT I - APP DEVELOPMENT – ANDROID/ FLUTTER/ FLASK						PW
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	40	60	1

Guidelines for Mini Project

1. The mini project is a team activity with a maximum of 3 students in a team. This is software-based design work.
2. The mini project may be a combination of hardware and software
3. Mini Project should cater to a small system required in laboratory or real life.
4. After interactions with course coordinator and based on comprehensive literature survey/ need analysis, the student shall identify the title and define the aim and objectives of mini project.
   a. Students are expected to detail specifications, methodology, resources required, critical issues involved in design and implementation and submit the proposal within the first week of the semester.
   b. The student is expected to exert on design, development and testing of the proposed work as per the schedule.
   c. Completed mini project and documentation in the form of mini project report is to be submitted at the end of semester.
5. The Mini Project will be evaluated for total of 100 marks where 40 marks are from CIE and 60 marks from SEE

Course Code	Course Title						Core / Elective
M24PC353AI	STATISTICAL ANALYTICS AND COMPUTING USING PYTHON LAB						PC
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Programming for Problem Solving using C Lab	-	-	-	2	40	60	1

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To introduce the fundamentals of Python programming and data types.
2. To enable students to implement control structures, functions, recursion, and string operations.
3. To facilitate learning of core data structures like lists, dictionaries, and sets in Python.
4. To develop the ability to read/write from files and manipulate data using libraries like NumPy and Pandas.
5. To equip students with skills for data analysis and visualization using Python.

COURSE OUTCOMES:

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

1. Apply basic Python programming concepts like variables, operators, and control structures to solve simple computational problems.
2. Analyze problems involving strings, recursion, and modularity using functions to write structured Python code.
3. Implement and evaluate programs using Python's built-in data structures like lists, dictionaries, and sets for solving data-centric problems.
4. Create and manipulate data using NumPy and Pandas for statistical computation and visualization.
5. Interpret data from various sources (files, CSVs) and visualize using Python libraries like matplotlib to derive meaningful insights.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3				2							1	2	
CO2	3	2			2							1	3	2
CO3	3	3	2		3							1	3	3
CO4	3	2	2		3							1	3	3
CO5	3	2	3		3							1	3	3

LIST OF PROGRAMS:

1. Write a program to declare variables and assign different types of data to those variables and display the data types of those variables.
2. Write a program to perform different Arithmetic operations on numbers in Python.
3. Write a program to find the area of a triangle whose sides are read by command line arguments.
4. Write a program to convert the temperature in degree centigrade to Fahrenheit.
5. Write a program to find the circumference and area of a circle with a given radius.
6. Write a program to check whether the given number is prime or not.

7. Write a program to check leap year.
8. Write a program to find the average of 10 numbers using while loop.
9. Write a program to display the given integer in a reverse manner.
10. Write a program to find the sum of the digits of an integer using a while loop.
11. Write a program to display all integers within the range 100-200 whose sum of digits is an even number.
12. Write a program to find the roots of a quadratic equation and display the nature of roots.
13. Write a program to find the factorial of a number using recursion.
14. Write a program to find the Nth term in a Fibonacci series using recursion.
15. Write a program to create, concatenate and print a string and access sub-string from a given string.
16. Write a program to check whether a string is palindrome or not.
17. Write a program to create a list and display the sum of list members.
18. Write a program to implement linear and binary search.
19. Write a program to find the largest number in a list without using built-in function.
20. Write a program to create a dictionary and print all the items in a dictionary.
21. Write a Program to create two sets and display union, intersection, difference, and symmetric difference of the two sets.
22. Write a python program to define a module and import a specific function in that module to another program.
23. Write a program to implement a calculator to do basic operations using functions.
24. Write a program that inputs a text file and prints all the unique words in the file in alphabetical order.
25. Write a program named copyfile.py. This file should prompt the user for the names of two text files. The contents of the first file should be copied.
26. Write a python program to a NumPy array and calculate the mean value, standard deviation, or maximum/minimum.
27. Write a python program to create two dimensional arrays using NumPy and perform matrix multiplication.
28. Create a DataFrame with pandas and display it.
29. Load a CSV file into a Pandas DataFrame and display a line plot, a bar plot, a histogram, box plot.

TEXT BOOKS

1. Introduction to Python Programming, Gowrishankar S., Veena A, CRC Press, Taylor & Francis Group, 2019.
2. Learning to Program with Python, Richard L. Halterman, Copyright © 2011.
3. A First Course in Probability and Statistics, B. L. S. Prakasa Rao,World Scientific/ Cambridge University Press India, 2009.

SKILL ENHANCENMENT COURSE

Guidelines for Evaluation of Skill Enhancement Course

1. Continuous Evaluation method is adopted for skill enhancement courses of all semesters and 40 marks are allocated for CIE.
   At the end of each module, the student is evaluated by allocating marks as given under.

Observation	: 10 marks
Continuous Performance and Execution	: 20 marks
Viva-Voce	: 10marks

   Average of marks obtained in all experiments is considered as the marks obtained in CIE

2. The Semester End Examination shall be conducted with an external examiner and the internal examiner for 60 marks. The external examiner shall be appointed by the Principal from the panel of examiners recommended by Controller of Evaluation and Board of Studies.

Quiz/ Skill Test/Assignment/ Mini Project	: 40 marks
Viva-Voce	: 20 marks

Course Code	Course Title				Core / Elective		
M24SE351AI	SKILL ENHANCENMENT COURSE-IOT				SE		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	40	60	1

COURSE OBJECTIVES:

The objective of this course is to make the student to

1. Learn how digital transformation turns information into action, creating unprecedented economic opportunity.
2. Understand how the IoT brings together operational technology and information technology systems.
3. Discover how business processes for evaluating and solving problems are being transformed.
4. Learn the security concerns that must be considered when implementing IoT solutions.
5. Practice what you learn using Cisco Packet Tracer, a network configuration simulation tool.

COURSE OUTCOMES:

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

1.Explain the meaning and impact of Digital Transformation.
2.Apply basic programming to support IoT devices.
3.Explain how data provides value to Digital Business and Society.
4.Explain the benefits of automation in the digitized world.
5.Explain the need for enhanced security in the digitized world and discover opportunities provided by digital transformation.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	1	2								2		
CO2	3	2	1									3		
CO3	3	2	3	2	2							3		
CO4	3	2	1	1								3	1	1
CO5	3	2	3	2	2							3	3	3

MODULE 1: Everything is Connected

Digital Transformation: Explain how digital transformation affects business, industry, and our daily lives, explain how digital transformation enables innovation, explain how networks provide the platform for Digital Business and society.

Devices that Connect to the IoT: Configure an IoT device to connect to the network, describe the exponential growth of connected IoT devices, configure devices to communicate in the IoT

MODULE 2: Everything Becomes Programmable

Apply Basic Programming to Support IoT Devices: Use Python to create programs that accept user input and read and write to external files, Describe basic programming variables and fundamentals. Apply basic programming variables and fundamentals in Blockly. Apply basic programming variables and fundamentals using Python

Prototyping Your Idea: Explain prototyping and its purpose,Describe Prototyping, Describe the various tools and materials to use to prototype.

MODULE 3: Everything Generates Data

Big Data: Explain the concept of Big Data, Describe the sources of Big Data, Explain the challenges and solutions to Big Data storage, Explain how Big Data analytics are used to support Business.

MODULE 4: Everything Can be Automated

What Can be Automated?: Explain how digitization allows business processes to embrace automation, Describe automation Explain how artificial intelligence and machine learning impact automation. Explain how intent-based networking adapts to changing business needs.

MODULE 5: Everything Needs to be Secured

Security in the Digitized World: Explain why security is important in the digitized world. Explain the need for security in the digitized world, explain how to help secure the corporate world, and explain how to secure personal data and devices.

REFERENCES

1. Introduction to IoT by CISCO Network Academy, Version 2.0,July 2018

IV SEMESTER
SYLLABUS

CSE(AI) Semester - IV									
S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination		Credits
			L	T	P / D	Contact Hours / week	CIE	SEE	
Theory Courses									
1	M24ES401EC	Logic Design and Computer Architecture	3	1	0	4	40	60	4
2	M24PC405AI	Software Engineering	3	0	0	3	40	60	3
3	M24PC406AI	Design and Analysis of Algorithms	3	0	0	3	40	60	3
4	M24PC407AI	Data Science	3	0	0	3	40	60	3
5	M24HS402HS	Human Values and Professional Ethics	2	0	0	2	40	60	2
6	M24PC408AI	MOOCS	0	0	0	0	0	0	3
Practical / Laboratory Courses									
7	M24PC454AI	Data Science Lab	0	0	2	2	40	60	1
8	M24PC455AI	Java Programming Lab	0	0	2*2	4	40	60	2
9	M24PW456AI	Mini Project I - App Development – Android/ Flutter/ Flask	0	0	2	2	40	60	1
		Total Credits				**23**	**320**	**480**	**22**

Course Code	Course Title						Core / Elective
M24ES401EC	LOGIC DESIGN AND COMPUTER ARCHITECTURE						ES
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	1	-	4	40	60	4

COURSE OBJECTIVES:

The objective of this course is to make the student

1. To understand basic number systems, logical gates, Boolean algebra & k map to minimize the Boolean expressions.
2. To understand design of combinational and sequential circuits
3. To learn the fundamentals of computer organization and its relevance to classical and modern problems of computer design.
4. To be familiarized with the hardware components and concepts related to the memory organization.
5. To be familiarized with the hardware components and instruction set related to 8086 microprocessor

COURSE OUTCOMES:

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

1. Explain binary number systems, Boolean algebra principles, computer organization components, memory hierarchies, and parallel processing architectures.
2. Apply Boolean algebra and K-map techniques to simplify logic expressions and design basic combinational circuits.
3. Demonstrate instruction execution cycles with addressing modes and analyze CPU register operations in 8086 architecture.
4. Compare memory technologies and I/O transfer modes with their performance characteristics.
5. Analyze pipelining challenges in processor architectures and evaluate multiprocessor organization trade-offs.

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2			1				1		2	2		1
CO2	3	2	1		1						2	2	1	1
CO3	3	3	1	1	1						2	3		1
CO4	3	3		2	1						2	3		
CO5	3	3	1	3					1		2	3		1

UNIT I

Binary Systems, Boolean algebra and Logic Gates: Number Base Conversions- Binary, Decimal, Octal and Hexadecimal, Complements – 1's Complement, 2's Complement.Digital Logic gates, Boolean algebra, Boolean Functions, Canonical and Standard Forms

Gate Level Minimization: The K Map Method, Four-Variable Map, Product of Sums Simplification. Don't-Care Conditions

UNIT II

Combinational Logic Design: NAND and NOR Implementation, Exclusive-OR Function, Design Procedure for Binary Adder, Subtractor, Decoders, Encoders, Multiplexer, Demultiplexer

Sequential Logic Design: Flip-Flops – SR, D, JK, T Flipflops.

Dept. of Computer Science
Methodist College of Engg. & Tech
King Koti, Hyderabad.

UNIT III

Basic Computer Organization: Functions of CPU, I/O Units, Memory Unit, Instruction Cycle
Central Processing Unit Organizations: General Register Organization, Stack Organization Instruction Formats- One address, two addresses, zero addresses and three addresses, addressing modes with numeric examples.

UNIT IV

Input-Output Organizations Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Direct Memory Access (DMA), Input-Output Processor (IOP), Intel 8089 IOP
Memory Organizations: Memory hierarchy, Main Memory, Associative Memory, Cache Memory, Miss and Hit ratio, Virtual memory

UNIT V

8086 Architecture: General purpose registers, Segment register, 8086 Flag register, Address Generation
Parallel Processing: Pipelining - Arithmetic Pipeline, Instruction Pipeline
Multiple Processor Organizations - Types of Parallel Processor Systems, Parallel Organizations, Symmetric Multiprocessors - Organization, Multiprocessor Operating System Design Considerations, A Mainframe SMP, Multicore Organization

TEXTBOOKS

1. Digital Design, M. Morris Mano, IV Edition, Pearson Education, Inc, 2002
2. Computer system architecture, M. Morris Mano. III Edition, Prentice-Hall, Inc., 1993.
3. Advanced Microprocessors and Peripherals Bhurchandi, Kishor M, III Edition, Tata McGraw Hill India, 2006.

REFERENCE BOOKS

1. Computer organization and architecture: designing for performance, William Stallings, VIII Edition, Pearson Education India, 2016.
2. Microprocessor 8086: Architecture, Programming and Interfacing, Sunil, Mathur. PHI Learning Pvt. Ltd., 2010.