MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(Deemed University)
(Declared Under Distinct Category by Ministry of Education, Government of India)
NAAC Accredited with A++ Grade
Department of Engineering Mathematics & Computing

Annexure-I

B.Tech. II SEM Scheme Admitted Batch 2025

Recommended in the BoS Meeting of "Department of Engineering Mathematics & Computing" held on 03rd December, 2025

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed University)

(Declared Under Distinct Category by Ministry of Education, Government of India)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics & Computing

Semester-Wise General Scheme Structure &Important Guidelines for Flexible Curriculum

(Batch admitted 2025 in Academic Session 2025-26 onwards)

Abbreviations Used

L	Lecture
T	Tutorial
P	Practical
HSMC	Humanities and Social Sciences including Management Courses
BSC	Basic Science Courses
ESC	Engineering Science Courses
DC	Departmental Core
DE	Departmental Elective
OC	Open Category
DLC	Departmental Laboratory Courses
MOOC	Massive Open Online Course
MWS	Mandatory Workshop
SP	Semester Proficiency
SIP	Skill Internship Program
SLP	Self-learning Presentation
PSC	Professional Skills & Competencies
PDC	Professional Development Component
PBL	Project Based Learning
PC	Professional Certification
MAC	Mandatory Audit Course
NEC	Novel Engaging Course

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed University)

(Declared Under Distinct Category by Ministry of Education, Government of India)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics & Computing

Scheme of Evaluation

B. Tech. II Semester *(MAC)*

(for 2025 batch admitted in academic session 2025-26)

S. No.	Course Code	Category Code	Course Name	Maximum Marks Allotted						Total Marks	Contact Hours per week			Total Credits	Mode of Learning	Mode of Major Evaluation	Duration of Major Evaluation
				Theory Block				Practical Block									
				Continuous Evaluation			Major Evaluation	Continuous Evaluation	Major Evaluation		L	T	P				
				Minor Evaluation I	Minor Evaluation II	Quiz/ Assignment		Lab Work & Sessional									
1.	25251201	DC	Object Oriented Programming & Methodology	25	25	20	30	-	-	100	3	-	-	3	Face to Face	MCQ	2 Hrs
2.	25251202	DC	Data Structures	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
3.	25251203	DC	Numerical Techniques	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
4.	25251204	DC	Computer Organization & Architecture	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
5.	25251205	DC	Linear Algebra	25	25	20	30	-	-	100	2	-	-	2	Face to Face	MCQ	2 Hrs
6.	25251206	DLC	Data Structures Lab	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
7.	25251207	DLC	Object Oriented Programming & Methodology Lab	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
8.	25251208	DLC	Numerical Computation using MATLAB	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
9.	25251209	SP	Semester Proficiency[$]	-	-	-	-	50	-	50	-	-	2	1	Face to Face	SO	-
10.	25251210	PBL	Micro Project-II[#]	-	-	-	-	70	30	100	-	-	2	1	Experiential	SO	-
11.	NECXXXXX	NEC	Novel Engaging Course (Activity Based Learning)	-	-	-	-	50	-	50	-	1	-	1	Interactive	SO	-
12.	SIP1XXXX	SIP	Skill Internship Program (Soft Skill)	-	-	-	-	60	-	60	-	-	-	2**	Experiential	SO	-
Total				**125**	**125**	**100**	**150**	**440**	**120**	**1060**	**11**	**04**	**10**	**22**	-	-	-
13.	25251211	MAC	Sustainability & Environmental Science	-	-	-	-	100	-	100	-	2	-	GRADE	Blended	SO	-
14.	25251212	MWS	Mandatory Workshop on Career Planning & Goal Setting at Department Level											GRADE	Interactive	MCQ	-

Summer Semester of six-eight week duration will be conducted for makeup of I & II semester examination.

[$]Semester Proficiency– includes the weightage towards ability/ skill/ competency /knowledge level /expertise attained etc. in the semester courses

MCQ: Multiple Choice Question **AO:** Assignment + Oral **PP:** Pen Paper **SO:** Submission + Oral **OB:** Open Book

** These credits will be transferred from Skill Internship Program (Soft Skill).

[#]Micro Project-II will be presented and evaluated through an interdisciplinary project evaluation committee.

HSMC	BSC	ESC	DC	DE	SPC	OC	DLC	NEC	SP	SIP	SLP	PDC	PBL	MAC	MWS
0	0	0	5	0	0	0	3	1	1	1	0	0	1	1	1

Mode of Learning						Mode of Examination					Total Credits
Theory		NEC	Lab			Theory				Lab	
Face to Face	Online	Interactive	Blended	Experiential	Experimental	PP	AO	MCQ	OB	SO	
15	0	1	0	3	3	7	3	7	0	5	22
68.18%	0	4.54%	0	13.64%	13.64%	31.81%	13.64%	31.81 %	0	22.72 %	Credits %

Recommended in the BoS Meeting of "Department of Engineering Mathematics & Computing" held on 03rd December, 2025

Department of Engineering Mathematics and Computing

Annexure II

Sem-II

Syllabus of B.Tech. –MAC (2025 Batch)

Department of Engineering Mathematics and Computing

Object Oriented Programming & Methodology

Subject Code: 25251201

Course Objectives

- To study about the concept of object-oriented programming.
- To create C++ programs that leverage the object-oriented features.
- To apply object-oriented techniques to solve real world problems.

Unit I:

Object Oriented Paradigm, Features of OOPs: Encapsulation, Class and Object, Inheritance, Reusability, Polymorphism, Abstraction etc, Comparison with Procedural Oriented Programming & Object-Oriented Programming, Function Overloading, Default Augments, References, Inline Functions.

Unit II:

Classes & Objects: Specification of Class, Visibility Modes: Private, Public, Protected, Defining Member Functions, Creating of Objects, Static Data Member, Static Member Function, Array of Objects, Object as Arguments, Friend Function and Class, Member Function, Member Initializer List, Constructors and Destructors, Deference between Class and Structure.

Unit III:

Dynamic Allocations: New, Delete, Malloc and Free, Dynamic Allocation of Objects, Array of Objects, Mutable Data Members, Self-Referential Class, Shallow and Deep Copying, This Pointer, Proxy Classes.
Operator Overloading: Overloading Unary and Binary Operators, **Type Casting:** Implicit, Explicit, Dynamic, Static, Reinterpret, Conversion Between Objects of Various Classes.

Unit IV:

Inheritance: Introduction to Code Reuse, Visibility Modes, Types of Inheritance, Ambiguity in Inheritance, Virtual Base Classes, Constructors in Derived Classes. **Polymorphism:** Dynamic and Static Binding, Pure Virtual Function, Abstract and Concrete Classes, Virtual Destructors, Containership: Nesting of Classes.
Exception Handling: Try, Catch and Throw, Streams and File: Basic Concept and Class Hierarchy

Unit V:

OOPs in Current Technologies and Languages: Java, Python, C#, and JavaScript, Kotlin and Swift, **Design Principles and Patterns**: SOLID Principles, Object Cloning and Metaprogramming, Dynamic OOP in Web Frameworks, etc.

Recommended Books

1. C++ How to Program: H M Deitel and P J Deitel, Prentice Hall,1998.
2. Object Oriented Programming in Turbo C++: Robert Lafore, The WAITE Group Press, 1994.
3. Programming with C++: D Ravichandran, T.M.H, 2003.
4. Object oriented Programming with C++: E Balagurusamy, Tata McGraw-Hill, 2001.
5. The Complete Reference in C++: Herbert Schildt, TMH, 2002.
6. Object Oriented Analysis & Design: G. Booch, Addison Wesley, 2006.
7. Principles of Object-Oriented Analysis and Design: James Martin, Prentice Hall, 1992.

Course Outcomes

After completion of this course, the students would be able to:

CO1. understand Fundamental concepts and features of Object-Oriented Programming such as classes, objects, encapsulation, inheritance, polymorphism, and abstraction.
CO2. apply OOP constructs including constructors, destructors, static members, friend functions, operator overloading, and dynamic memory allocation to develop modular programs.
CO3. discuss different types of inheritance, visibility modes, binding mechanisms, and object-copying techniques to determine their suitability in object-oriented design.
CO4. demonstrate concept of polymorphism, abstract classes, exception handling, and file handling to create robust and reusable OOP-based applications.
CO5. design & develop modern OOP concepts, principles and contemporary OOP tools to scalable simple applications in current technologies.

Department of Engineering Mathematics and Computing

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Data Structures
Subject Code: 25251202

COURSE OBJECTIVES

- To be familiar with the use of data structures as the foundational base for computer solutions to problems.
- To understand various techniques of searching and sorting.
- To understand basic concepts about stacks, queues, lists, trees and graphs.

Unit I:
Introduction to Data Structures: Algorithms & their Characteristics, Asymptotic Notations, Arrays and its Representations, Index to Address Translation, **Link List:** Introduction, Implementation of Linked List, Operations, Circular Link List, Doubly Linked List, Polynomial Manipulation using Linked List.

Unit II:
Stacks: Concepts and Implementation of Stacks, Operations on Stack, Conversion of Infix to Postfix Notation, Evaluation of Postfix Expression, Recursion.
Queues: Concepts and Implementation, Operations on Queues, Dequeue, Priority Queues, Circular Queue.

Unit III:
Trees: Types, Terminology, Binary Tree -Representations, Traversal, Threaded Binary Tree, Binary Search Tree, Height Balanced Tree-AVL Tree.
Graph: Terminologies, Representation of Graphs- Sequential & Linked Representation, Graph Traversals- BFS, DFS, Spanning Trees.

Unit IV:
Searching: Linear Search, Binary Search, Hashing and Collision Resolution Techniques; **Sorting:** Bubble Sort, Selection Sort, Insertion Sort.

Unit V:
Advanced Data Structures: Graph-Based Data Structures such as k-d Trees, R-Trees, Hashing Techniques for Databases & Storage such as Extensible Hashing, Cuckoo Hashing, Perfect Hashing, etc.

Recommended Books

1. Data Structures, Algorithms and Applications in C++, Sartaj Sahni, 2nd Edition.
2. An Introduction to Data Structures with Applications, Jean-Paul Tremblay, Mcgraw hill.
3. Data Structures & Algorithms, Aho, Hopcroft & Ullman, Original Edition, Pearson Publication.

Course Outcomes
After completion of this course, the students would be able to:
CO1. explain the concept of algorithms, linked list data structures.
CO2. apply the appropriate data structure stack and queue to solve problems.
CO3. discuss the concept of tree and graph data structure, types & their applications.
CO4. design various searching and sorting algorithms and analyze their performance.
CO5. discover the applications of data structure in emerging areas and real world.
CO6.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Numerical Techniques
(Subject Code: 25251203)

Course Objective

- To perceive the Error of computation
- To enumerate the concept of interpolation, extrapolation, numerical differential, and integration
- To understand numerical solution of ordinary differential equation
- To explore the finite difference methods

Unit 1:

Problem solving on computer, Algorithms and flow charts, Introduction to numerical computing, approximations and errors in numerical computations. Useful rules for estimating Errors, Truncation and round off errors, propagation of errors, Error in the Approximation of function, Error in Approximation

Bisection method, Regula Falsi method, Iteration method, Newton Raphson method, Secant method, convergence of iterative methods.

Unit 2:

Matrix algebra, Solution of simultaneous linear algebraic equations: Gauss elimination, Gauss Jordan method, LU decomposition, Jacobi method, Gauss Seidel method, SOR method, Ill and well condition of equations, Condition of a system and stability issues., Finite Differences, forward, backward and central operators, Shifting operators, Averaging Operators, Differences of a polynomial, Factorial Notation, Relation between operators.

Unit 3:

Newton's forward and backward interpolation formula, Lagrange interpolation formula, Divided differences and Newton's divided difference formula, Inverse Interpolation, Numerical differentiation, Numerical integration: Newton-Cotes integration formulas, Trapezoidal, Simpson's rules (1/3 & 3/8) and Weddle rules.

Unit 4:

Taylor series method, Picard's method, Euler's method, Modified Euler's method, RungeKutta methods fourth order. Multistep methods: Milne's Predictor corrector method, Numerical solution of the simultaneous linear differential equation, Second order differential equation.

Unit 5:

Classification of partial differential equation, Finite difference method, Numerical Solution of Partial Differential equations, five-point formula, Laplace and Poisson equation.

Course Outcomes

After completing this course, the students will be able to:

CO's	Description of CO's
CO1	Identify the concepts Algebraic & Transcendental Equations
CO2	Acquire the knowledge of difference operators
CO3	Illustrate numerical integration and differentiation
CO4	Evaluate the problems of ordinary differential equation employing numerical techniques
CO5	Test the Partial differential equations by finite difference method

Recommended Books:

1. B. S. Grewal: Higher Engineering Mathematics, Khanna Publisher, 43rd Edition, 2015.
2. B.V. Ramanna: Higher Engineering Mathematics, McGraw Hill,1st Edition, 2017.
3. S.S. Sastry: Introductory Methods of Numerical Analysis, PHI Learning Private Limited, 4th edition, 2007.
4. J. H. Mathews and K. D. Fink: Numerical Methods using MATLAB, PHI, 4th edition, 2007.
5. C.F. Gerald and P.O. Wheatley: Applied Numerical Analysis, Pearson Education, 6th edition, 2006.
6. H. K. Dass: Advance Engineering Mathematics, S. Chand& Company, Publisher, 2018.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Computer Organization and Architecture
Subject Code: 25251204

Course Objective

- To understand how the computer systems, work and its basic principles.
- To know about ALU operations, fixed point arithmetic and floating-point Arithmetic, instruction set architecture, execution cycle and different types of Control Units.
- To be familiar with the Memory System Design, different types of Memories.
- To understand how the I/O devices are accessed, its principles and concepts of pipelining and Parallel Processing.

Unit I:

Functional Blocks of a Computer: CPU, Memory, Input-Output Unit, Control Unit, Basic Operational Concepts, Von Neumann Architecture.

Data Representation: Signed Number Representation, Fixed- and Floating-Point Representations, Computer Arithmetic– Integer Addition and Subtraction, Ripple Carry Adder, Carry Look-Ahead Adder, Multiplication: Shift and Add, Booth Multiplier.

Unit II:

Introduction to 8086 Microprocessor: Architecture, Register Transfer Language, Register Transfer, Memory Transfer, Instruction Cycle, Addressing Modes, Instruction Set, CISC v/s RISC Architecture, CPU Control Unit Design: Hardwired and Micro-Programmed.

Unit III:

Memory Organization: Memory Hierarchy, Memory Interleaving, Cache Memory, Mapping Functions, Write Policies.

Peripheral Devices: Characteristics, I/O Device Interface, Data Transfer Modes, I/O Transfers: Program Controlled, Interrupt Driven and DMA, Interrupts and Exceptions.

Unit IV:

Pipelining: Basic Concepts of Pipelining, Pipelining Hazards, Parallel Processors: Introduction, Shared Memory Multiprocessors and Cache Coherency.

Unit V:

Emerging Architectures and Advanced Computing Paradigms: Introduction to GPUs and their Architecture, Comparison of CPU vs GPU, Instruction-Level Parallelism (ILP) and its importance in modern processors, Quantum Computing Basics, Neuromorphic Computing.

Recommended Books

1. Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB /McGraw-Hill
2. Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.
3. Computer System Design and Architecture", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

Course Outcomes

After completion of this course, the students would be able to:

CO1. Define the basics of computer systems, ALU operations, fixed point arithmetic and floating-point Arithmetic.
CO2. Explain different types of instructions and instruction execution cycle along with types of control units.
CO3. Illustrate the memory system design and different types of memories.
CO4. Explain the access of I/O devices and its principles.
CO5. Discuss the concepts of pipelining and parallel processing.

Course Articulation Matrix

Department of Engineering Mathematics and Computing

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Linear Algebra
Subject Code: 25251205

Objective of Course

- To understand the concept Matrices and its applications
- To comprehend the various aspect of algebraic structures
- To explore vector space
- To perceive knowledge of linear transformation and their application

Unit1:
Matrix, Rank of Matrix, Echelon form, Normal form of matrix, Solution of simultaneous equation by elementary transformation, Consistency of equation, Eigen values and Eigenvectors, Normalized eigenvector, Cayley Hamilton theorem and its application to finding inverse of matrix.

Unit2:
Introduction of Groups and its properties, Sub-groups, Coset, Lagrange's theorem for finite group, Ring and its properties, Field, Integral domain.

Unit3:
Vector spaces over the field and its properties, sub-spaces, linear dependent vectors and linear independent vectors, linear combination of vectors, linear span of a set of vectors, basis and dimension of a vector space.

Unit4:
Linear transformation, Kernel and range space of linear transformation, Nullity and Rank, Singular and Non-Singular transformation, Matrix representation of a linear transformation.

Unit5:
Inner product spaces, Properties of inner product space, Schwarz's inequality, Triangular inequality, Parallelogram Law, Orthogonality, Pythagoras theorem.

CO's	Description of CO's
CO1	Determine the solution of matrices
CO2	Find the analytical solution of algebraic structures
CO3	Relate the use of vector space in computation
CO4	Acquire the knowledge of linear transformations
CO5	Illustrate the concept of inner product spaces

Recommended Books:

1. S. Lipschutz and M. Lipson, Linear Algebra (4th Edition), Schaum's Outline series, Mc- Graw Hill. (2009).
2. S. Boyd and L. Vandenberghe, Introduction to Applied Linear Algebra Vectors, Matrices, and Least Squares, University Printing House, Cambridge CB2 8BS, United Kingdom One Liberty Plaza, 20thFloor, New York, NY10006, USA, (2018).
3. E.Kreyszig: Advance Engineering Mathematics, JohnWiley&Sons,10thEdition(2011).
4. R. K. Jain, S. R. K. Iyengar: Advance Engineering Mathematics, Narosa Publishing House Pvt. Ltd, 5th Edition (2016).

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Course Name: Sustainability & Environmental Science
Course Code: 25251211

L	T	P	Credit
0	2	0	GRADE

Course Objective:

To equip students with a comprehensive understanding of environmental science, pollution control, sustainability, and global frameworks, enabling them to analyze environmental challenges and contribute to sustainable solutions through informed decision-making and responsible practices.

Unit I:

Introduction to Environmental Science: definition, importance and its components. Ecosystem and its components. Water cycle, carbon cycle, food chain, energy flow in ecosystem. Current state of environment in India and world; Underlying reasons (root causes) of modern environmental degradation (social, psychological, cultural). Introduction to Environmental pollution: air, water, noise, soil, thermal and radioactive.

Unit II:

Environmental Pollution and Management: air, water, noise, soil, thermal and radioactive. Causes, impacts, pollution control techniques and mitigation strategies. Solid waste management: Principles of waste management, different components of waste management system and introduction to management of hazardous waste like e-waste, plastic waste. Global environmental Issues: Climate change, global warming, ozone layer depletion, urban heat island

Unit III:

Environmental policies and laws in India: Environmental Protection Act, Water Act, Air Act. **Overview of global environmental policies and frameworks:** Kyoto protocol, Montreal protocol, COP summits. Introduction to clean development mechanism, carbon credit, carbon trading. Environmental audit.

Unit IV:

Sustainability concepts: definition, importance, pillars of sustainability (economic, environmental, and social). Sustainable development. Overview of UN Sustainable Development Goals (SDGs) and their global relevance. Concept of circular economy, resource efficiency, energy conservation, green buildings and sustainable manufacturing.

Unit V:

Sustainable Energy solutions: New energy sources: need of new sources, different types of new energy sources, application of hydrogen energy, ocean energy sources, and tidal energy conversion. Concept, origin and power plant of geothermal energy. Renewable energy sources like water, wind etc. Overview of sustainable materials and construction practices. Introduction to sustainable transportation systems and sustainable water infrastructure.

Recommended Books:

1. D. K. Asthana, Meera Asthana, A Text Book of Environmental Studies, S Chand & Co., New Delhi.
2. S. K. Dhameja, Environmental Engineering & Management, S K Kataria & Sons, New Delhi
3. C. S. Rao, Environmental Pollution Control Engineering, C.S. Rao, New Age International Publishers
4. A. K. Gupta, Environmental Sustainability and Green Technologies, PHI Learning.

Course Outcomes:

Upon completion of the course, a student will be able to

CO1. **Explain** the fundamental concepts of environmental science, including ecosystems and the causes of environmental degradation.

CO2. Analyze the sources, causes, and impacts of air, water, and solid waste pollution and propose appropriate mitigation strategies.

CO3. Evaluate the effectiveness of environmental policies and global frameworks in addressing environmental challenges.

CO4. Explain the concepts of sustainability and sustainable development goals.

CO5. Apply various solutions for achieving sustainable development.

Department of Engineering Mathematics and Computing

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Annexure III

Sem-II

Experiment List/ Lab Manual (2025 Batch)

Department of Engineering Mathematics and Computing

Data Structures Lab
Subject Code: 25251206

LIST OF PROGRAMS

1. Write a program to implement doubly linked list with all possible deletion operations.
2. Write a program to insert an element in the beginning of the circular linked list.
3. Write a program to implement stack using linked list.
4. Write a program to count the number of nodes in the binary search tree.
5. Write a program to implement AVL Tree.
6. Write a program to traverse the BST in pre-order and post-order.
7. Write a program to implement Graph using an array.
8. Write a program to implement Breadth First Search.
9. Write a program to implement Depth First Search.
10. Write a program to implement Spanning Tree.
11. Write a program to implement binary search algorithm.
12. Write a program to implement Heap Sort.
13. Write a program for implementing the Radix Sort methods to arrange a list of integers in ascending order.
14. Write a program for implementing the Quick Sort methods to arrange a list of integers in ascending order.
15. Implement a simple Bloom Filter in C++ that supports insertion of string(s) and checks possibly contain string(s) using multiple hash functions.
16. Implement a Count–Min Sketch to process a stream of strings and answer approximate frequency queries.

Course Outcomes

After completion of the course students would be able to:

CO1. implement data structures like arrays and linked lists to manage and manipulate data effectively.
CO2. develop stack and queue operations, including applications like expression evaluation and recursion.
CO3. construct and traverse various tree structures, including binary search trees and height-balanced trees.
CO4. apply graph traversal techniques like BFS and DFS for solving pathfinding and connectivity problems.
CO5. design efficient sorting and searching algorithms, and resolve hashing collisions.

Course Articulation Matrix

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

Department of Engineering Mathematics and Computing

Object Oriented Programming & Methodology Lab
Subject Code: 25251207

LIST OF PROGRAMS

1. Write a program to demonstrate example of member initializer list.
2. Write a program to demonstrate example of default constructor or no argument constructor.
3. Write a program to demonstrate example of parameterized constructor.
4. Write a program to demonstrate example of copy constructor.
5. Write a program to demonstrate example of constructor overloading.
6. Write a program to demonstrate example of destructors.
7. Write a program to demonstrate example of constructor using this pointer.
8. Write a program to demonstrate example of constructor with default arguments.
9. Write a program to dynamic Initialization of Objects.
10. Write a program to set values of data members using default, parameterized and copy constructor
11. Write a program to demonstrate example of simple inheritance.
12. Write a program to demonstrate example of private simple inheritance.
13. Write a program to read and print student's information using two classes and simple inheritance.
14. Write a program to demonstrate example of multilevel inheritance.
15. Write a program to read and print employee information using multiple inheritance.
16. Write a program to demonstrate example of multiple inheritance.
17. Write a program to demonstrate example of hierarchical inheritance to get square and cube of a number.
18. Write a program to read and print employee information with department and PF information using hierarchical inheritance.
19. Write a program for unary minus (-) operator overloading.
20. Write a program for unary increment (++) and decrement (--) operator overloading.
21. Write a program for unary logical NOT operator overloading.
22. Write a program to add two objects using binary plus (+) operator overloading.
23. Write a program to add two distances using binary plus (+) operator overloading.
24. Write a program to create a simple class and object.
25. Write a program to create an object of a class and access class attributes.
26. Write a program to create multiple objects of a class.
27. Write a program to create class methods.
28. Write a program to define a class method outside the class definition.

Department of Engineering Mathematics and Computing

29. Write a program to assign values to the private data members without using constructor.
30. Write a program to create an empty class (a class without data members and member functions).
31. Write a program to create a class with setter and getter methods.
32. Write a program to create a class to read and add two distances.
33. Write a program to create a class for student to get and print details of a student.
34. Write a program to create a class for student to get and print details of N students.
35. Write a program to demonstrate example of array of objects.
36. Write a program to create class to read and add two times.
37. Write a program to create class to read time in seconds and convert into time in (HH:MM:SS).
38. Write a program to create class to read time in HH:MM:SS format and display into seconds.
39. Write a program to demonstrate example of friend function with class.
40. Write a program to count the created objects using static member function.
41. Write a program to create an object of a class inside another class declaration.
42. Write a program to create a class Point having X and Y Axis with getter and setter functions.
43. Write a program for passing an object to a Non-Member function.
44. Write a program for accessing Member Function by pointer.
45. Write a program for accessing the address of an object using 'this' pointer.
46. Write a program to create a class with public data members only.
47. Write a program to input list of candidates and find winner of the Election based on received votes.
48. Write a program for Banking Management System using Class.
49. Write a program to create a file.
50. Write a program to read a text file.
51. Write a program to write and read text in/from file.
52. Write a program to write and read values using variables in/from file.
53. Write a program to Build a simple calculator API using OOP and C++ web framework.
54. Create a program class and expose it through a web API.

Course Outcomes

After completion of the course students would be able to:

CO1. use the concepts of object-oriented programming, including classes, objects, constructors, destructors, and initializer lists.
CO2. apply OOP principles to develop programs using operator overloading and friend functions.
CO3. analyze relationships among classes through inheritance, object interaction, and dynamic initialization.
CO4. demonstrate object-oriented program structure using file handling, static members, and pointers.
CO5. design OOP-based applications, including web-exposed C++ classes and APIs.

Course Articulation Matrix

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

Department of Engineering Mathematics and Computing

CO5	3	3	3	2	2	1	1	1	1	3	2	3	1

Numerical Computation Lab

Subject Code: 25251208

Numerical Computation Using MATLAB
List of Topics in Experiments

1. Introduction to MATLAB and Simple Calculations with MATLAB.
2. Creating Arrays and Mathematical Operations in MATLAB.
3. Two Dimensional Plots in MATLAB.
4. User Define function and function file.
5. Loops and Conditional Statements in MATLAB.
6. Polynomial and Interpolation.
7. Application on Numerical Methods:

a. Solving Algebraic Equations of one variable
b. Finding Maxima & Minima
c. Numerical Integration.
d. Ordinary Differential Equation.

8. Three dimensional Plots.

List of Experiments

1. If X=[1 4; 8 3], Find
   - the inverse matrix of X.
   - the diagonal of X.
   - the sum of each column and the sum of whole matrix X.
   - the transpose of X.
2. Plot Sinc function in MATLAB, where Sinc (x) = sin(x) / x, and $-2\pi \le x \le 2\pi$
3. 3-D Plot of function: y=xcos(x); z=exp(x/5)cos(x)+1 for $0 \le x \le 6\pi$.
4. Root Finding
   - Program for roots of f(x)=0 by Newton Raphson method
   - Program for roots of f(x)=0 by Bisection method
   - Program for roots of f(x)=0 by Regula-Falsi method.
5. Solution of a system of simultaneous algebraic equations using the Gaussian Elimination procedure.
6. Determination of Eigenvalues and Eigenvectors of a square matrix.
7. Solution of a system of simultaneous algebraic equations using the Gauss-Seidel iterative method.
8. Program for solving to integral of a given function using Trapezoidal Rule
9. Program for solving numerical integration by Simpson's 1/3 rule. 10.Program for solving numerical integration by Simpson's 3/8 rule.
11. Program for solving numerical solution of an ordinary differential equation using the Euler's method.
12. Program for solving numerical solution of an ordinary differential equation using the Runge-Kutta -4th order method.

Course Articulation Matrix

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**
CO1	**2**	**1**	**-**	**-**	**-**	**-**	**1**	**-**	**-**	**1**	**1**	**-**	**-**
CO2	**2**	**2**	**2**	**-**	**-**	**-**	**3**	**-**	**-**	**1**	**2**	**-**	**-**
CO3	**1**	**1**	**1**	**-**	**-**	**2**	**2**	**-**	**-**	**1**	**2**	**-**	**-**
CO4	**1**	**1**	**-**	**-**	**-**	**-**	**2**	**-**	**-**	**1**	**2**	**-**	**-**

CO5	**2**	**2**	**2**	**-**	**-**	**1**	**3**	**-**	**-**	**1**	**2**	**-**	**-**

13. 1 - Slightly; 2 - Moderately; 3 – Substantially

B. Tech. (Second Semester)

Micro Project-II

25241210

List of Project

1. Tumor Growth Prediction Using Reaction-Diffusion Equations in MATLAB.
2. Euler's Method for Solving First-Order ODEs
3. Runge-Kutta Method (RK4) for Solving Differential Equations
4. Comparison of Root Finding Methods: Bisection vs Newton-Raphson
5. Simulation of Epidemic Spread Using SIR Model and Euler's Method
6. Simulation of Projectile Motion with Air Resistance Using Runge-Kutta Method
7. 1D Heat Equation using Implicit Crank-Nicolson Method and Five-point method
8. Edge Detection using Canny and Sobel Filters
9. Image Compression using Singular Value Decomposition (SVD)
10. Face Detection using Viola-Jones Algorithm
11. Object Tracking in a Video using Optical Flow
12. Weather Data Analysis using Principal Component Analysis (PCA)
13. Implementing Newton-Raphson Method for Root Finding
14. Population Growth Prediction using Logistic Regression
15. Image Segmentation using K-means Clustering
16. Histogram Equalization for Image Enhancement
17. Watermarking an Image using Discrete Wavelet Transform (DWT)
18. Morphological Operations for Noise Removal
19. Image Filtering using Gaussian and Median Filters
20. Handwritten Digit Recognition using Neural Networks
21. Speech Recognition using Mel-Frequency Cepstral Coefficients (MFCC)
22. Noise Reduction in Audio Signals using Low-pass Filtering
23. Frequency Analysis of a Musical Note using Fast Fourier Transform (FFT)
24. Signal Modulation and Demodulation (AM/FM)
25. Denoising ECG Signals using Wavelet Transform
26. Audio Steganography for Secure Communication
27. Beat Detection in Music using Spectrogram Analysis
28. Voice Gender Classification using Machine Learning

Department of Engineering Mathematics and Computing

29. Heart Rate Estimation from ECG Signals
30. Sound Localization using Microphone Arrays
31. Handwritten Digit Classification using SVM
32. Predicting Stock Prices using Linear Regression
33. Spam Email Detection using Naïve Bayes Classifier
34. Image Classification using Convolutional Neural Networks (CNN)
35. Credit Card Fraud Detection using Decision Trees
36. Sentiment Analysis of Twitter Data using NLP
37. Disease Prediction using K-Nearest Neighbors (KNN)
38. House Price Prediction using Random Forest
39. Traffic Sign Recognition using Deep Learning
40. Weather Forecasting using Time Series Analysis
41. PID Controller Design for a DC Motor
42. Inverted Pendulum Stabilization using State-Space Control
43. Obstacle Avoidance for a Mobile Robot using Fuzzy Logic
44. Path Planning using A* Algorithm for Autonomous Vehicles
45. Cruise Control System Simulation using MATLAB Simulink
46. Solving Ordinary Differential Equations (ODEs) using MATLAB Solver

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(Deemed University)
(Declared Under Distinct Category by Ministry of Education, Government of India)
NAAC Accredited with A++ Grade
Department of Engineering Mathematics & Computing

Annexure-IV

B.Tech. IV SEM Scheme Admitted Batch 2024

Recommended in the BoS Meeting of "Department of Engineering Mathematics & Computing" held on 03rd December, 2025

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed University)

(Declared Under Distinct Category by Ministry of Education, Government of India)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics & Computing

Semester-Wise General Scheme Structure &Important Guidelines for Flexible Curriculum

(Batch admitted in Academic Session 2025-26 onwards)

Abbreviations Used

L	Lecture
T	Tutorial
P	Practical
HSMC	Humanities and Social Sciences including Management Courses
BSC	Basic Science Courses
ESC	Engineering Science Courses
DC	Departmental Core
DE	Departmental Elective
OC	Open Category
DLC	Departmental Laboratory Courses
MOOC	Massive Open Online Course
MWS	Mandatory Workshop
SP	Semester Proficiency
SIP	Skill Internship Program
SLP	Self-learning Presentation
PSC	Professional Skills & Competencies
PDC	Professional Development Component
PBL	Project Based Learning
PC	Professional Certification
MAC	Mandatory Audit Course
NEC	Novel Engaging Course

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed University)

(Declared Under Distinct Category by Ministry of Education, Government of India)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics & Computing

Scheme of Evaluation

B. Tech. IV Semester *(MAC)*

(for 2024 batch admitted in academic session 2025-26)

S. No.	Course Code	Category Code	Course Name	Maximum Marks Allotted						Total Marks	Contact Hours per week			Total Credits	Mode of Learning	Mode of Major Evaluation	Duration of Major Evaluation
				Theory Block				Practical Block									
				Continuous Evaluation			Major Evaluation	Continuous Evaluation	Major Evaluation		L	T	P				
				Minor Evaluation I	Minor Evaluation II	Quiz/ Assignment		Lab Work & Sessional									
1.	25242201	DC	Data Science	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
2.	25242202	DC	Database Management System	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
3.	25242203	DC	Software Engineering	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
4.	25242204	BSC	Transforms and Vector Calculus	25	25	20	30	-	-	100	2	1	-	3	Face to Face	MCQ	2 Hrs
5.	25242205	DC	Number Theory and Cryptography	25	25	20	30	-	-	100	3	-	-	3	Face to Face	MCQ	2 Hrs
6.	25242206	DLC	Data Science Lab	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
7.	25242207	DLC	Database Management System Lab	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
8.	25242208	DLC	Competitive Programming	-	-	-	-	70	30	100	-	-	2	1	Experimental	AO	-
9.	25242209	SP	Semester Proficiency[$]	-	-	-	-	50	-	50	-	-	2	1	Face to Face	SO	-
10.	25242210	PBL	Macro Project-II[#]	-	-	-	-	70	30	100	-	-	2	1	Experiential	SO	-
11.	NECXXXXX	NEC	Novel Engaging Course (Activity Based Learning)	-	-	-	-	50	-	50	-	1	-	1	Interactive	SO	-
12.	SIP3XXXX	SIP	Skill Internship Program (Evaluation)	-	-	-	-	60	-	60	-	-	-	2**	Experiential	SO	-
Total				125	125	100	150	440	120	1060	11	05	10	23	-	-	-
13.	25242211	MAC	Project Management, Economics & Financing	-	-	-	-	100	-	100	-	2	-	GRADE	Blended	SO	-
14.	25242212	MWS	Mandatory Workshop on Intellectual Property Rights at Department Level											GRADE	Interactive	MCQ	-

Summer Semester of six-eight week duration will be conducted for makeup of previous semester examination.

Additional Course for Honours or Minor Degree: Permitted to opt for maximum two additional courses for the award of Honours or Minor Degree

[$]Semester Proficiency– includes the weightage towards ability/ skill/ competency /knowledge level /expertise attained etc. in the semester courses

MCQ: Multiple Choice Question **AO:** Assignment + Oral **PP:** Pen Paper **SO:** Submission + Oral **OB:** Open Book

[#] Macro Project-II will be presented and evaluated through an interdisciplinary project evaluation committee. [**] These credits will be transferred from Skill Internship Programme.

PC	BSC	ESC	DC	DE	SPC	OC	DLC	NEC	SP	SIP	SLP	PDC	PBL	MAC	MWS
0	1	0	4	0	0	0	3	1	1	1	0	0	1	1	1

Mode of Learning						Mode of Examination					Total Credits
Theory		NEC	Lab			Theory				Lab	
Face to Face	Online	Interactive	Blended	Experiential	Experimental	PP	AO	MCQ	OB	SO	
16	0	1	1	3	3	7.5	3	7.5	0	5	23
66.66 %	0	4.17 %	4.17 %	12.5 %	12.5 %	32.60%	12.5 %	32.60 %	0	21.74 %	Credits %

Recommended in the BoS Meeting of "Department of Engineering Mathematics & Computing" held on 03rd December, 2025

Department of Engineering Mathematics and Computing

Annexure-V

Sem-IV

Syllabus of B.Tech. –MAC (2024 Batch)

Department of Engineering Mathematics and Computing

Transforms and Vector Calculus (25242204)

Objective of Course:

- To perceive the transform techniques in engineering problems
- To expose the concept of Fourier series and Fourier Transform
- To understand Wavelet transform and Z-Transform
- To explore the Vector Calculus

Unit 1: **Fourier Series**

Introduction to Fourier Series, Dirichlet's conditions, Parseval's Identity, Fourier series for arbitrary and periodic functions, Fourier series for even and odd functions, Half range Fourier series, and its properties, Complex form of Fourier series and Harmonic analysis.

Unit 2: **Fourier Transforms**

Fourier integral, Complex Fourier transform, linear property, change of scale property, Shifting property, Fourier transform of derivative of a functions, inverse Fourier Transforms, Convolution theorems, Fourier sine and cosine transform, applications of Fourier transform to differential equations.

Unit 3: **Laplace Transform & Its Applications**

Definition of Laplace Transform, conditions for existence of Laplace Transform. Properties of Laplace transform,

Unit step functions, Dirac delta-function. Inverse Laplace transform, convolution theorem, Solution of ordinary differentialequations with the initial and boundary conditions.

Unit 4: **Z- Transform and Difference Equations**

Introduction to Z- transform, Properties of the Z-Transform, Inverse Z-Transform, Convolution, Partial Fraction Method, Residual Method and Solving Linear Difference Equation Using Z-Transform.

Unit 5: **Vector Calculus**

Introduction of Vector calculus, Vector differentiation: Gradient, Divergence, and Curl, directional derivative, Solenoidal and Irrotational vectors, Vector Integration: Applications for finding Length, surface and Volume using multiple integrals.

Course Outcomes

After completing this course, the students will be able to:

CO's	Description of CO's
CO1	**Develop** half-range Fourier sine and cosine series and use their properties to represent functions defined on limited intervals.
CO2	**Compute** complex Fourier transforms, sine/cosine transforms, and inverse Fourier transforms for various classes of functions.
CO3	**Explain** the definition of the Laplace transform, conditions for its existence, and the importance of Laplace techniques in engineering.
CO4	**Solve** linear difference equations using the Z-transform technique for discrete-time engineering problems.
CO5	**Evaluate** line, surface, and volume integrals of vector fields and apply them to engineering and physical problems.

Recommended Books:

1. B. S. Grewal: Higher Engineering Mathematics, Khanna Publisher, 43rd Edition, 2015.
2. G. ShankerRao: Mathematical Methods, I. K. International Publications, 1st Edition, 2009.
3. Ian N. Sneddon: Fourier Transforms, Dover Publications, 2010.
4. Loknath Debnath: Integral Transforms and their applications, Chapman and Hall/CRC, 2nd edition, 2006.
5. Narayan Shanti and P. K. Mittal: A Text Book of Vector Analysis, S. Chand, Company, 2010 Edition.

Department of Engineering Mathematics and Computing

Course Articulation Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	2	1	1	1	1	1	1	3	3	3
CO2	3	3	3	3	2	1	1	1	1	1	1	3	3	3
CO3	3	3	3	3	2	1	1	1	1	1	1	3	3	3
CO4	3	3	3	3	2	1	1	1	1	1	1	3	3	3
CO5	3	3	3	3	2	1	1	1	1	1	1	3	3	3

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Database Management Systems (25242202)

Objective of Course:

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modelling, relational, hierarchical and network models.
- To understand and use data manipulation language to query, update and manage a database.

Unit I:
DBMS: Database Approach v/s Traditional File Approach, Advantages of Database System, Database Users and Administrator, Database System Environment, Application Architectures, Schemas, Instances, Data Independence, Data Models: Hierarchical Data Model, Network Data Model & Relational Data Model, Comparison between Models. Entities and Relationship Model: Entity types, Entity sets, Attributes and Keys, Relationship Types and Sets, Constraints, Design issue, E-R Diagram, Weak Entity Sets.

Unit II:
Relational Model: Structure of Relational Databases: Relation, Attribute, Domain, Tuples, Degree, Cardinality, Views, Database Relations, Properties of Relations, Attributes, Keys, Attributes of Relation, Domain Constraints, Integrity Constraints. Relational Algebra: Concepts and Operations: Select, Project, Division, Intersection, Union, Division, Rename, Join etc.

Unit III:
SQL: Purpose of SQL, Data Definition Language (DDL) Statements, Data Manipulation Language (DML) Statements Update Statements & Views in SQL, Data Control Language (DCL), Triggers. Relational Database Design: Purpose of Normalization, Data Redundancy and Update Anomalies, Functional Dependency, Process of Normalization, Various Normal Forms: 1NF, 2NF, 3NF, BCNF, Decomposition, Desirable Properties of Decomposition: Dependency Preservation, Lossless Join, Problems with Null Valued & Dangling Tuple, Multivalued Dependencies.

Unit IV:
Transaction Management: Transaction Concept, Transaction State, Concurrent Executions, Serializability: Conflict and View Serializability, Concurrency Control: Lock-Based Protocol, Recovery: Log-Based Recovery.

Unit V:
Big Data and Distributed Databases: CAP theorem, eventual consistency, sharding. NoSQL and NewSQL Databases: Document (MongoDB), Key-Value (Redis), Column-family (Cassandra), Graph (Neo4j), Google Spanner, CockroachDB. Real-Time Data Processing and Streaming Databases: Apache Kafka, Apache Flink.

Recommended Books:

1. Database System Concepts, Abraham Silberschatz Henry F. Korth S. Sudarshan, McGraw-Hill 6th Edition.
2. Database Management System, Raghu Ramakrishnan Johannes Gehrke, McGraw Hill 3rd Edition.
3. Fundamentals of Database System, Elmasri & Navathe, Addison-Wesley Publishing, 5th Edition.
4. An Introduction to Database Systems, Date C. J, Addison-Wesley Publishing, 8th Edition.

Department of Engineering Mathematics and Computing

Course Outcomes

After completion of this course, the students would be able to:

CO's	Description of CO's
CO1	differentiate between traditional file systems and the database approach.
CO2	construct relational schemas and demonstrate the use of relational algebra operations to query and manipulate relational data
CO3	develop SQL queries for data definition, manipulation, and control, and analyze relational schemas for normalization.
CO4	examine transaction processing concepts and analyze concurrency control and recovery mechanisms in database systems.
CO5	compare traditional RDBMS with modern distributed and NoSQL databases, and evaluate their suitability for real-time data processing scenarios.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Number Theory and Cryptography (25242205)

Objective of Course:

- To Understand the Crypto graphical techniques to converting some secret information to not readable texts
- Explore the Crypto graphical techniques in various applications such as include military information transmission, computer passwords, electronic commerce, and others.
- Introduce the idea of encryption and public key cryptosystem in the context of algebra and elementary number theory.

Unit 1:

Number theory, Divisibility theory, Modular Arithmetic, primes and their distribution, theory of congruence and its applicationin security, Congruence: basic definitions and properties, complete and reduced residue systems.

Unit 2:

Integer representations (binary and base expansions, base conversion algorithm), Fermat's Little Theorem and Euler's Theorem,primitive roots, quadratic reciprocity, and Divisibility: basic definition, properties, prime numbers, some results on distribution of primes.

Unit 3:

Arithmetical functions: examples, with some properties and their rate of growth; Continued fractions, and their connections with Diophantine approximations, applications to linear and Pell's equations; Binary quadratic forms; Partition: basic properties and results; Diophatine equations: linear and quadratic, some general equations.

Unit 4:

Overview of cryptography, Encryption, Symmetric Encryption, Plain text, cipher text, Historical Ciphers, Shift Cipher, Substitution Cipher, Vigen'ere Cipher, Permutation Cipher, Symmetric Ciphers, Stream Cipher, Block Ciphers. Symmetric Key Distribution, key management, secret key distribution, public and private key cryptography.

Unit 5:

RSA cryptosystem, Primality Testing and Factoring, Key Exchange and Signature Schemes Diffie–Hellman Key Exchange, Digital Signature Schemes, Cryptographic hash functions, Authentication, Digital Signatures, Identification, certification, Discrete logarithm problem in general and on finite fields. Polynomials on finite fields, irreducibility and their applications to coding theory.

Course Outcomes

After completing this course, student will be able to:

CO's	Description of CO's
CO1	Acquire the knowledge of number theory and transcendental numbers
CO2	Describe the divisibility and related algorithms, factorization and quadratic sieve, efficiency of other factoring algorithms.
CO3	Evaluate arithmetical functions, Distribution of primes and Diophantine equations
CO4	Apply cryptography tools in various applications
CO5	Examine the Public key cryptosystems

Recommended Books:

1. Nigel Smart : Cryptography : An Introduction, CRC Press,3rd edition, 2013
2. Neal Koblitz : A course in number theory and cryptography, Springer-Varlag, 2nd edition, 1994.
3. W. Stein: Elementary Number Theory: Primes, Congruences and Secrets, OPAQUE, 2017
4. Burton, David M. Elementary Number Theory, 7th ed., 2011, McGraw-Hill, Inc.
5. Koshy, Thomas. Elementary Number Theory With Applications, 2nd ed., 2007. Elsevier, Inc
6. Robbins, Neville. Beginning Number Theory, 1993. Iowa: Wm. C. Brown
7. P. S. Gill, Cryptography and Network Security, 2011, Oxford Publication
8. Stein, William, Elementary Number Theory: Primes, Congruences, and Secrets: A Computational Approach, 2017,Springer Verlag

Department of Engineering Mathematics and Computing

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Software Engineering (25242203)

Objective of Course

- To understand the nature of software development and software life cycle process models, agile software development, SCRUM and other agile practices.
- To understand project management and risk management associated with various types of projects.
- To know basics of testing and understanding concept of software quality assurance and software configuration management process.

Unit: 1

Introduction to Software Engineering: Definition, software engineering-layered Technology, Software Characteristics and Components, Software model: Software Development of Life Cycle Model (SDLC), The Waterfall Model, Iterative Waterfall Model, Prototyping Model, Spiral Model, RAD Model. Selection criteria of model: Characteristics of Requirements, Status of Development Team, Users participation, Type of Project and Associated Risk.

Unit: 2

Requirement Engineering: Definition, Requirement Engineering Activity, Types of Requirements- Functional and Non-functional Requirements, User and System Requirements, Requirement Elicitation Methods, Requirement Analysis Methods, Requirement Documentation (SRS), Requirement Validation, Requirement Management.

Unit: 3

Design Concept, Principle and Methods: Design Fundamentals, Design Principles, Effective Modular Design, Design Representations, Architectural design, Procedural design, data Directed design, Real Time Design, Object Oriented Design, Coupling and Cohesion.

Unit: 4

Software Metrics, Project Management and Estimation: Metrics in Process and Project domains, Software Measurement, Software Quality Metrics, Project Management- Basics-People, Product, Process, Project, Estimation- Software Project Estimation, Decomposition Techniques- Function Point Estimation, Line of Code (LOC) based estimation, Empirical Estimation, COCOMO Model, Project Scheduling Techniques.

Unit: 5

Software Testing: Definitions, Software Testing Life Cycle (STLC), Test Case Design, Strategic Approach to Software Testing-Verification & Validation , Strategic issues, Criteria for completion of Testing, Unit Testing, Integration Testing, Validation Testing, System Testing, Black Box Testing Techniques, White Box Testing Techniques, Acceptance Testing.

Course Outcomes

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

CO's	Description of CO's
CO1	Explain the various fundamental concepts of software engineering
CO2	Develop the concepts related to software design & analysis.
CO3	Compare the techniques for software project management & estimation
CO4	Choose the appropriate model for real life software project.
CO5	Test the software through different approaches.

RECOMMENDED BOOKS:

1. Software Engineering: A Practitioner's Approach, Roger S. Pressman, McGraw Hill, 2001.
2. Software Engineering, K.K. Agrawal & Yogesh Singh, New Age Publication, 2007.
3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2014.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Data Science (25242201)

Objective of Course:

- To enable students to acquire, preprocess, analyze, and visualize data, and perform exploratory and statistical analysis for data-driven decision-making.
- To develop skills in building, evaluating, and validating predictive models using traditional techniques and modern data tools.

Unit I:

Introduction to Data Science: Scope, lifecycle, evolution, and real-life applications, Types of Data: Structured, Unstructured and Semi Structured Data, Data Formats: TXT, CSV, JSON, XLSX, Reading and writing files.

Unit II:

Data Acquisition, Pre-Processing, and Visualization: Data Collection Strategies, Data Pre-Processing: Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization, Data Normalization Data Visualization Techniques: Box Plot, Scatter Plot, Histogram, Bar Chart, and Pie Chart, Outlier Detection.

Unit III:

Exploratory Data Analytics: Descriptive Statistics: Measures of central tendency: Mean, Median, Mode, Measures of dispersion: Variance, Standard Deviation, Shape of distribution: Skewness and Kurtosis, Correlation Analysis: Computing correlation coefficients between features, visualizing correlations using heatmaps, Hypothesis Testing: t-test, Chi-square test, ANOVA.

Unit IV:

Model Development and Evaluation: Model Development: Regression Models: Simple Regression, Multiple Regression, Classification Models: Logistic Regression, Decision Tree, Random Forest. Performance Evaluation Measures: Regression and Classification, Model Validation: Train-Test split, Cross-Validation, Learning Curves, Detecting Under fitting and Overfitting.

Unit V:

Ethical Data Management & Tools for Analytics and Visualization: Data Quality Management and Data Governance: Data Privacy, Security, and Ethical Handling, Data Science Tools: Introduction to Open Refine, Trifacta, KNIME, Tableau, Power BI, Google Data Studio, Airtable, Zapier, Power Automate for data preparation, visualization, and workflow automation.

RECOMMENDED BOOKS

1. McKinney, W. (2018). *Python for data analysis*. O'Reilly Media.
2. Kelleher, J. D. (2018). *Data science*. MIT Press.
3. Blum, A., Hopcroft, J., & Kannan, R. (2020). *Foundations of data science*. Cambridge University Press.
4. Bruce, P. (2019). *Data science: What the best data scientists know*. Wiley.
5. Park, A. (2020). *Data science: A beginner's guide*. Independently Published.
6. Singh, N., & Huo, K. (2021). *Ace the data science interview*. Data Interview Pro.

Department of Engineering Mathematics and Computing

Course Outcomes

After completing the course, the student will be able to:

CO's	Description of CO's
CO1	understand data science concepts, data types, formats, and applications.
CO2	apply data acquisition, preprocessing, normalization, and visualization techniques for real datasets.
CO3	perform exploratory data analysis using descriptive statistics, correlations, and hypothesis testing methods.
CO4	develop and evaluate regression and classification models using appropriate validation techniques.
CO5	implement ethical data handling and perform workflows using modern no-code platforms.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Course Code: 25242211

Course Name: Project Management, Economics & Financing

L	T	P	Credit
0	2	0	GRADE

Course Objectives:

To provide knowledge about project attributes and planning essentials, develop project networks, make rational decisions for project completion, utilize resources effectively, and understand the basics of project finances and management.

Unit I:

Project Planning: Introduction to Project Management, Difference between Project and Production, Attributes of a Project: Time, Cost, Quality and Safety. Stakeholders of a Project, Project life cycle. Project Planning: Types of Project Plans and feasibility.

Unit II:

Project Network logic: Project Networking and work flows, Activity duration and methods of estimating activity duration – One time estimate three time estimates, Duration estimation procedure. Use of Bar Charts, Mile stone charts and networks, Network representation schemes: Activity on Arrow and Activity on Node Networks (A-o-A & A-o-N), Logic behind developing project network and simple network calculations, Critical paths and floats.

Unit III:

Decision making through networks: CPM, PERT & PDM: Use of network in Decision Making: Importance of critical path, Monitoring the progress and updating the project plan. Use of floats in Resource smoothening, Introduction to Precedence Diagramming Method (PDM), Different lag and lead relations in terms of SS(Start to Start), SF(Start to Finish), Finish to Start(FS), and Finish to Finish(FF) and composite relations.

Unit IV:

Project Cost Control: Breakeven analysis in planning stage, Direct and indirect cost, slope of direct cost curve, Total project cost and optimum duration, contracting the network for cost optimization. Escalation & Variation in prices.

Unit V:

Projects Financing: Introduction to project financing; Role of governments in financing projects, Funder and Concessionaire: Economic multiplier effects of Projects; Means of financing-public finance and private finance, Granting authority: World Bank Group, IMF,ADB, Micro and Small Enterprises Funding Scheme (MSME), Elementary understanding of Procurement of infrastructure projects through Public Private Partnership (PPP) route, Build Operate Transfer (BOT), Build Operate Own & Transfer (BOOT); Stakeholders' perspectives, Lifecycle of PPP projects, Micro & Macro economics concepts and its application in Project Financing.

Course Outcomes: At the end of the course student will be able to

CO 1: Know the attributes of project and its different phases.
CO 2: Develop the project network based on work breakdown structure and estimation of activity durations.
CO 3: Analyze the project network and make decide the various alternates.
CO 4: Evaluate the optimum cost of project for assigned deadlines.

Department of Engineering Mathematics and Computing

CO 5: Understand the different options to arrange the finances to complete it within stipulated time.

Text-Books:

1. Project Management Scheduling PERT and CPM by Dr. B.C. Punmia, K.K. Khandelawal

2. PERT & CPM Principles and Applications by L.S. Srinath, Affiliated EWP Pvt. Ltd.

3. Project Planning and Control by Albert Lester, Fourth Edition Elsevier Butterworth-Heinemann.

Reference Books:

1. A Management Guide to PERT/CPM With GERT/PDM/DCPM and Other networks by Jerome D. Wiest, Ferdinand K. Levy, Prentice Hall.

2. Project Management with CPM and PERT by Joseph J . Moder, Cecil R . Phillips, Van Nostrand Reinhold Company

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Annexure-V

Sem-IV

LAB Syllabus of B.Tech. –MAC-IV SEM

Database Management System Lab (25242207)

List of Programs

While creating tables and databases, ensure that each name begins with your roll number as a prefix. Example: If your roll number is 25, then every table name should start with "25_TABLE_NAME".
Additionally, follow the instructions below:

- *Mention the program name clearly.*
- *Provide a brief description of each command used to execute the query.*
- *Write all SQL commands in bold letters.*
- *Attach a screenshot of the output.*

1. Implementation of DDL Commands in SQL

- Create table (CREATE TABLE)
- Modify table structure (ALTER TABLE)
- Delete table (DROP TABLE)

2. Implementation of DML Commands in SQL

- Insert records (INSERT INTO)
- Update records (UPDATE)
- Delete records (DELETE)

3. Implementation of SQL Functions

- Number Functions
- Aggregate Functions
- Character Functions
- Conversion Functions
- Date Functions

4. Implementation of SQL Operators

- Arithmetic Operators
- Logical Operators
- Comparison Operators
- Set Operators
- Special Operators (LIKE, BETWEEN, IN)

5. Implementation of SQL Joins

- Inner Join
- Outer Join
- Natural Join

6. Grouping, Filtering & Indexing

- GROUP BY Clause
- HAVING Clause
- ORDER BY Clause
- Creating Index (CREATE INDEX)

7. Subqueries and Views

- Single-row & Multi-row Subqueries
- Creating and Managing Views (CREATE VIEW, DROP VIEW)

8. Implementation of Constraints

- Primary Key
- Foreign Key
- Unique
- Not Null
- Check
- Default

9. Database Backup & Recovery

- Export & Import Operations
- Backup Commands
- Recovery Operations

10. Transaction Control Language (TCL)

- COMMIT
- ROLLBACK
- SAVEPOINT

11. Creating Database / Table Space

- CREATE DATABASE
- Managing Storage Structure

12. User & Role Management

- Create User
- Drop User
- Grant Privileges
- Revoke Privileges

13. Trigger Implementation

- BEFORE and AFTER Triggers
- Row-level and Statement-level Triggers

14. ACID Properties & Transaction Management

- Atomicity
- Consistency
- Isolation
- Durability
- Practical Execution with SQL Queries

15. MongoDB NoSQL Operations

- CRUD Operations
- Aggregation Pipeline
- Collections & Documents
- Basic Queries in MongoDB Compass or Shell

COURSE OUTCOMES LAB:

After completion of this course lab, the students would be able to:

CO1. understand the fundamental concepts of SQL, including data types, DDL, DML, DCL, and TCL commands.
CO2. enable to design, create, and manage databases using constraints, functions, operators, and advanced SQL features.
CO3. develop skill proficiency in writing SQL queries involving joins, subqueries, grouping, indexing, triggers, and transaction control.
CO4. utilize database objects such as Tables and Views in appropriate scenarios.
CO5. develop a mini-project demonstrating end-to-end database design and implementation for a real-world application.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Data Science LAB (25242206)

List of Programs

1. Write a program that reads datasets from TXT, CSV, JSON, and XLSX formats, converts a CSV file into JSON and a JSON file into XLSX, writes the cleaned data into a selected folder.
2. Apply complete data preprocessing on the Adult Income Dataset and Loan Prediction Dataset, including Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization, and Data Normalization.
3. Write a program to write a data frame given in Fig.1 to csv file and analyse a given sample dataset by finding its data distribution using histogram, column chart, box plot, scatter plots, pie chart identifying outliers through visualization, and creating histogram, bar chart, and pie chart to represent the data.

S. No.	Empid	Gender	Age	Sales	BMI	Income
1	E1	M	32	123	NORMAL	350
2	E2	M	35	114	OVERWEIGHT	450
3	E3	F	34	135	OBESITY	160
4	E4	F	42	139	UNDERWEIGHT	190
5	E5	F	26	117	NORMAL	90
6	E6	M	38	121	OBESITY	140
7	E7	F	44	133	OVERWEIGHT	125
8	E8	M	24	140	UNDERWEIGHT	95
9	E9	M	36	133	OVERWEIGHT	200
10	E10	F	40	133	OBESITY	310

Fig.1

4. Perform EDA on Credit Card Fraud Detection Dataset (open-source dataset) for analysing the data.
5. Apply Regression Model techniques to predict the future values of data on the open-source available datasets. (Housing Price Prediction Dataset, Stock Market Prices Dataset)
6. Import the Iris Flower dataset from the UCI Machine Learning Repository containing three species of iris flowers (Setosa, Versicolor, Virginica). Apply a logistic regression model for multi-class classification to categorize the iris samples based on their features.
7. Plot Accuracy and Error Metrics against number of iterations for evaluation of model performance (consider any dataset mentioned in program 2-6).
8. Plot Learning curves for model evaluation for Under-fitting and Over-fitting (consider any dataset mentioned in program 2-6).
9. Write a Python script to clean a CSV file by removing duplicates and normalizing column names, similar to OpenRefine operations.
10. Write code to automate reading a CSV file, transforming selected columns, and exporting a summary report, replicating a KNIME or Power Automate workflow.

Department of Engineering Mathematics and Computing

Course Outcomes (COs)

CO1. **CO1**: apply techniques to read, convert, clean, and manage diverse data.

CO2. implement pre-processing methods to prepare datasets for analysis tasks.

CO3. perform exploratory data analysis and visualize data distributions using various plots.

CO4. implement predictive models for classification and regression to interpret the data.

CO5. create automated data workflows and pre processing scripts inspired by no-code tools.

Course Articulation Matrix

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

1 - Slightly; 2 - Moderately; 3 – Substantially

Department of Engineering Mathematics and Computing

Data Science:List of skill-based Mini Project

- Exploratory Data Analysis (EDA): Perform an in-depth analysis of a dataset, including data cleaning, visualization, and statistical analysis to gain insights and understand the underlying patterns and relationships.
- Predictive Modeling: Build a machine learning model to predict a specific outcome or target variable based on a given dataset. This could include classification, regression, or time series forecasting tasks.
- Natural Language Processing (NLP): Develop a text classification or sentiment analysis model using techniques such as tokenization, word embeddings, and recurrent neural networks (RNNs) to analyze and understand text data.
- Image Recognition: Create an image recognition system using convolutional neural networks (CNNs) to classify or identify objects, faces, or patterns in images.
- Recommendation System: Build a recommendation engine that suggests personalized recommendations to users based on their preferences and behavior, using collaborative filtering or content-based filtering techniques.
- Clustering Analysis: Implement clustering algorithms such as k-means, hierarchical clustering, or DBSCAN to group similar data points together and discover hidden patterns or segments within a dataset.
- Time Series Analysis: Analyze time-dependent data, such as stock prices or weather data, using techniques like autoregressive integrated moving average (ARIMA), exponential smoothing, or recurrent neural networks (RNNs).
- Anomaly Detection: Develop an anomaly detection system that can identify unusual or suspicious patterns in data, which can be useful for fraud detection, network intrusion detection, or outlier detection.
- Social Media Sentiment Analysis: Use data from social media platforms to analyze public sentiment towards specific topics, brands, or events using natural language processing techniques and sentiment analysis algorithms.
- Data Visualization Dashboard: Create an interactive dashboard using libraries like Plotly or Dash to visualize and explore data, providing users with an intuitive interface to interact with and gain insights from the data.

Department of Engineering Mathematics and Computing

Course Code: 25242208

Course Name: Competitive Programming

Course Objectives:-

- To build strong foundational skills in programming logic, time–space complexity, and problem-solving techniques used in competitive programming.
- To enable students to apply data structures, algorithms, graphs, dynamic programming, and number theory concepts for solving medium to advanced-level coding problems.
- To prepare students for national and global coding contests by developing speed, accuracy, debugging skills, and consistent practice on online judge platforms.

UNIT 1: Basics of C++/Python, input–output, loops, conditionals, functions. Introduction to time–space complexity and constraints. Arrays, strings, frequency maps, simple logic problems. Basics of STL (vector, set, map, pair). Practice: 40–50 beginner problems on LeetCode/CodeChef.

UNIT 2: Core Data Structures & Techniques: Stacks, queues, linked lists, heaps, hash maps. Two-pointer, sliding window, prefix sums, hashing. Binary search and sorting-based logic patterns. Practice on GFG DS Track & LeetCode/CodeChef Data Structures. Target: 30–40 DS-based problems.

UNIT 3: Graphs & Dynamic Programming, Graph basics: BFS, DFS, components, cycles. Shortest paths: Dijkstra, Bellman-Ford, MST algorithms. Recursion, memoization, DP introduction. DP topics: knapsack, LIS, LCS, grid DP.Practice: 30-40 graph & DP problems on LeetCode/CodeChef.

UNIT 4: Advanced Algorithms & Number Theory, Sieve, modular arithmetic, combinatorics, modular inverse. Fast exponentiation, prime factorization. Greedy strategies and optimization techniques. Bit manipulation & bitmask DP. Practice: 30–40 advanced-level problems across platforms.

UNIT 5: Contest Practice & Deployment Registration on LeetCode, CodeChef, GFG, HackerRank. Weekly contests, mock tests, debugging strategies. Editorial writing and clean coding practices. Mini-project: CP notebook/GitHub repository. Target: 30–40 problems + 6 mock contests.

Course Outcomes (COs)

CO1: Solve beginner-level CP problems using programming fundamentals.
CO2:Apply appropriate data structures and algorithmic techniques for intermediate problems.
CO3: Use graph algorithms and dynamic programming to solve complex problems.
CO4:Implement advanced number theory, greedy methods, and optimization in contest environments.
CO5: Participate effectively in coding contests and maintain consistent problem-solving practice.

Reference Books

1. "Competitive Programming 4" by Steven Halim & Felix Halim (2018).
2. "Guide to Competitive Programming" by Antti Laaksonen (Springer, 2017).
3. "Programming Challenges: The Competitive Programmer's Handbook" by Laaksonen (Revised Edition 2016).
4. Elements of Programming Interviews in C++" by Adnan Aziz et al. (2016).

Course Articulation Matrix

Department of Engineering Mathematics and Computing

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

1 - Slightly; 2 - Moderately; 3 – Substantially

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed to be University)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics and Computing

Annexure-VII

SCHEME
(Under Flexible Curriculum)

B. Tech. Mathematics and Computing with Honors and Minors

Admitted Batch (2024)

MAC IV SEM.

Madhav Institute of Technology & Science, Gwalior-474005

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(Deemed to be University)
NAAC Accredited with A++ Grade

Department of Engineering Mathematics and Computing

B. Tech. IV Semester (Honors 2024 Batch)
[Total 18 credits to be opted in the program]

S.No	Subject Code	Subject Name	Theory MOOC			Credits
			End Sem.	Assignment /Quiz	Total Marks	
1.		Artificial Intelligence: Search Methods for Problem solving	75	25	100	03
2.		Applied Accelerated Artificial Intelligence	75	25	100	03
3.		Introduction to Machine Learning	75	25	100	03

B. Tech. IV Semester (Minors 2024 Batch)

S.No	Subject Code	Subject Name	Theory MOOC			Credits
			End Sem.	Assignment /Quiz	Total Marks	
1.		Differential Equations	75	25	100	03
2.		Partial Differential Equations for Engineers	75	25	100	03
3.		Applied Linear Algebra in AI and ML	75	25	100	03

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(Deemed University)
(Declared Under Distinct Category by Ministry of Education, Government of India)
NAAC Accredited with A++ Grade
Department of Engineering Mathematics & Computing

ANNEXURE - VIII

B.Tech. MAC-IV

List of Professional Certification Platforms and Certification

Session 2025-26

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(Deemed University)

(Declared Under Distinct Category by Ministry of Education, Government of India)

NAAC Accredited with A++ Grade

Department of Engineering Mathematics & Computing

List of Professional Certification Platforms and Certification

S.No.	Course Name	Duration (weeks/month /hours)	Platform	Web Link
1	Mathematics for Machine Learning and Data Science Specialization	3 month	Coursera	https://www.coursera.org/specializations/mathematics-for-machine-learning-and-data-science
2	Microsoft Power BI Data Analyst	5 months	Coursera	https://www.coursera.org/professional-certificates/microsoft-power-bi-data-analyst
3	Full stack web development and AI with Python (Django)	40 Hours	Udemy	https://www.udemy.com/course/unaicorn/?srsltid=AfmBOoqJzmZvlp645-yc9O94mH-phRLIPNIHGi3o4-ZUVNfXlvjES5DS&utm_source=chatgpt.com&couponCode=LEARNNOWPLANS
4	Google Data Analytics Professional Certificate	6 months	Coursera	https://www.coursera.org/professional-certificates/google-data-analytics
5	Applied Generative AI Specialization	16 Weeks	Simplilearn	https://www.simplilearn.com/applied-ai-course?source=preview_AI%20%26%20Machine%20Learning_card&eventname=Mega_Menu_New_Select_Category_card
6	Google IT Automation with Python Professional Certificate	6 Months	Coursera	https://www.coursera.org/professional-certificates/google-it-automation

- Students can also be enrolled in other such courses from verified platform.
- The Credit of professional certification will be transferred in VIth Sem.