Course Curriculum (Scheme & Syllabi)

for

B. Tech.

in

Artificial Intelligence & Data Engineering

AY 2026-27

DEPARTMENT OF ARTIFICIAL INTELLIGENCE & DATA ENGINEERING

MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

Tentative Scheme and Syllabi

First Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
1	26MAT101	Mathematics I	2-1-0	3	***IC***
2	26PHT151	Physics	2-1-0	3	***IC***
3	26CST101	Programming with Python	2-0-0	3	***IC***
4		Environmental Science	1-0-0	1	***IC***
5	26HSP101	Soft Skill Foundational Lab	0-0-2	1	***IC***
6	26HST101	Human Values and Professional Ethics	1-0-0	1	***IC***
7		Engg. Drawing and Sketching Lab	0-0-2	1	***IC***
8	26PHP101	Physics Lab	0-0-2	1	***IC***
9	26AIT101	Data Structure using C	3-0-0	3	**PC**
10	26AIP101	Data Structure using C Lab	0-0-2	1	**PC**
11	26AIT102	Computational Perception and Cognition	2-0-0	2	**PC**
			Total	**20**	

Second Semester					
S. No	**Code**	**Subject**	**L-T-P**	**Credit**	**Type**
1	26MAT151	Mathematics II	2-1-0	3	IC
2	26ICP101	Makers Lab	0-3-6	6	IC
3	26CHT101	Data Analytics	2-0-0	2	IC
4	26AIT151	Introduction to Artificial Intelligence	3-0-0	3	PC
5	26AIT152	Mathematics for AIDE	3-0-0	3	PC
6		Signals and Systems	3-0-0	3	PL(EAS/BS)
			Total	20	

Malaviya National Institute of Technology Jaipur
Department of Artificial intelligence and data engineering

Course Details
Data Structure using C | 26AIT101 | 3 0 0 0

PREREQUISITE:
NIL

COURSE OBJECTIVE(S):
The course aims to develop a strong foundation in C programming, data structures, and algorithmic problem solving. It covers linear and non-linear data structures, searching, sorting, hashing, trees, heaps, and graphs along with their time and space complexity. Students learn to select, analyze, and apply suitable data structures and algorithms for solving computational problems efficiently.

COURSE OUTCOMES:

CO1 : Explain C programming fundamentals, problem-solving techniques, recursion, and dynamic memory management.

CO2 : Analyze and implement linear data structures such as arrays, linked lists, stacks, queues, and deques.

CO3 : Apply searching, sorting, and hashing techniques and compare their efficiency.

CO4 : Explain and analyze trees, AVL trees, heaps, priority queues, B-trees, and B+ trees.

CO5 : Represent graphs and apply traversal, shortest-path, and minimum-spanning-tree algorithms using suitable data structures.

SYLLABUS:

UNIT I (8 Lectures): Introduction to computers, computational thinking, problem-solving steps, algorithms, flowcharts, fundamentals in C, data types, storage classes, operators, expressions, formatted input/output and command-line arguments. Control and Functions : Decision and looping statements, functions, parameter passing, scope, recursion

UNIT II (8 Lectures): Concept and classification of data structures, memory allocation, dynamic memory allocation using malloc(), calloc(), realloc() and free(), abstract data types, asymptotic notation and analysis of time and space complexity. Linear Data Structures in C: Arrays and linked lists (singly, doubly and circular), stack representation, operations and applications, queue implementation, sequential and circular queues, deque, dynamic queue implementation and applications. C structures, pointers and dynamic memory are used throughout the implementations. basic file handling using C.

UNIT III (8 Lectures): Linear search and binary search, bubble, selection, insertion, merge, quick, counting, bucket, radix and heap sorting, stability, in-place behavior and comparison of sorting algorithms based on complexity and use cases. Hashing and Hash Tables: Hash functions, open and closed hashing, collision handling through chaining and probing, dynamic and extendible hashing, design and implementation of hash tables in C.

UNIT IV (10 Lectures): Binary-tree terminology and representations, tree traversals, binary search trees, threaded binary trees, list representation and dynamic implementation of binary trees, AVL trees and rotations, tree applications, interval trees, M-way search trees, B-trees and B+ trees. Heaps: Heap representation and operations, priority queues, heap applications and C-based implementations of tree and heap operations.

UNIT V (8 Lectures): Graph fundamentals, directed and undirected graphs, adjacency matrix and adjacency-list representations in C, graph traversal using depth-first search and breadth-first search, connected components and basic applications. Graph Algorithms: Shortest-path computation using Dijkstras algorithm, minimum spanning trees using Prims algorithm, implementation considerations, complexity analysis and comparative problem solving using suitable data structures.

EVALUATION

Final grades will be weighted average of performance in

Course Work	20%
Midterm Exam	30%
Endterm Exam	50%

depending on the institute's academic policy.

Textbook(s):

Reference book(s):

1. Tenenbaum, A. M., Langsam, Y., and Augenstein, M. J. Data Structures Using C. Pearson Education.
2. Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. Introduction to Algorithms, 3rd ed. MIT Press, 2009.
3. Kernighan, B. W., and Ritchie, D. M. The C Programming Language, 2nd ed. Prentice Hall.
4. Aho, A. V., Hopcroft, J. E., and Ullman, J. D. Data Structures and Algorithms. Addison-Wesley.
5. Levitin, A. Introduction to the Design and Analysis of Algorithms, 2nd ed. Pearson.

Online/E-Resourses:

Malaviya National Institute of Technology Jaipur
Department of Artificial intelligence and data engineering

Course Details
Computational Perception and Cognition | 26AIT102 | 2 0 0 0

PREREQUISITE:
NIL

COURSE OBJECTIVE(S):
The course aims to introduce computational and cognitive approaches for understanding the mind, perception, representation, and intelligence. It explores symbolic and connectionist models, philosophical debates in AI, and the relationship between cognition and computation. Students are also introduced to modern connectionist architectures such as RNNs, attention mechanisms, transformers, and LLMs.

COURSE OUTCOMES:

CO1 : Explain philosophical, psychological, and cognitive approaches to modeling the mind and representation

CO2 : Describe the role of computation and AI in theories of cognition.

CO3 : Analyze symbolic representations and models used to explain cognitive processes.

CO4 : Evaluate major debates and limitations in modeling intelligence, including functionalism and machine thinking.

CO5 : Explain connectionist models, perception, RNNs, attention, transformers, and LLMs in relation to cognition and intelligence.

SYLLABUS:

UNIT I (4 Lectures): Philosophical, Psychological, and Cognitive approaches to modeling the mind, ingredients of computation and nature of representation, approaches to characterising, formalising, and modelling the mind, role of intentionality

UNIT II (4 Lectures): The relevance of computation and AI, types of computation, physical symbol system hypothesis

UNIT III (4 Lectures): Symbolic representations and models of cognition

UNIT IV (8 Lectures): Broader philosophical debates, challenges and limitations in modelling the mind, functionalism and the rise of AI The First AI Debate: Can Machines Think?

UNIT IV (8 Lectures): RNNs, attention, transformers, LLMs, implications for cognitive science and notions of intelligence, knowing, and understanding

EVALUATION
Final grades will be weighted average of performance in

Course Work	20%
Midterm Exam	30%
Endterm Exam	50%

depending on the institute's academic policy.

Textbook(s):

Reference book(s):

1. Jay Friedenberg and Gordon Silverman. Cognitive Science: An Introduction to the Study of Mind (3rd ed.). SAGE, 2015.
2. Cameron J. Buckner. From Deep Learning to Rational Machines: What the History of Philosophy Can Teach Us about the Future of Artificial Intelligence. Oxford, 2023.

Online/E-Resourses:

Course Details
Data Structure using C Lab | 26AIP101 | 0 0 2 0

PREREQUISITE:
NIL

COURSE OBJECTIVE(S):
The laboratory course aims to develop practical programming skills for implementing data structures and algorithms using C. Students gain hands-on experience with linked lists, stacks, queues, sorting, searching, hashing, trees, heaps, and graphs. The course also strengthens debugging, dynamic memory management, and implementation of algorithms such as DFS, BFS, Dijkstra's, and Prim's

COURSE OUTCOMES:
CO1 : Develop and debug C programs using arrays, structures, functions, pointers, recursion, and dynamic memory allocation.
CO2 : Implement linear data structures such as linked lists, stacks, queues, and deques in C.
CO3 : Implement and compare searching, sorting, and hashing techniques using C.
CO4 : Implement trees, AVL trees, heaps, priority queues, B-trees, and B+ trees.
CO5 : Represent graphs and implement DFS, BFS, Dijkstra's, and Prim's algorithms in C.

SYLLABUS:

EXPERIMENT I (6 Hours): C Programming Fundamentals and Problem Solving: Programs using input/output, operators, decision and loop constructs, arrays, structures, functions, pointers, recursion, and dynamic memory allocation. Practice in compilation, debugging, and modular program development

EXPERIMENT II (6 Hours): Implementation of arrays and sparse matrices, singly, doubly, and circular linked lists, stacks using arrays and linked lists, linear queues, circular queues, deques, and their applications.

EXPERIMENT III (6 Hours): Programs for linear search, binary search, and hashing with chaining and probing. Implementation and comparison of bubble, selection, insertion, merge, quick, counting/radix, and heap sorting methods.

EXPERIMENT IV (6 Hours): Binary-tree creation and traversals, binary search tree operations, AVL-tree rotations, threaded-tree concepts, binary heaps and priority queues, introductory implementation of B-tree and B+ tree operations.

EXPERIMENT V (6 Hours): Graph representation using adjacency matrix and adjacency list, graph traversal using DFS and BFS, implementation of Dijkstra shortest-path and Prim minimum-spanning-tree algorithms.

EVALUATION
Final grades will be weighted average of performance in

Course Work	60%
Midterm Exam	20%
Endterm Exam	20%

depending on the institute's academic policy.

Textbook(s):

Reference book(s):

1. Tenenbaum, A. M., Langsam, Y., and Augenstein, M. J., Data Structures Using C, Pearson Education
2. Kernighan, B. W., and Ritchie, D. M., The C Programming Language, 2nd Edition, Prentice Hall.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C., Introduction to Algorithms, 3rd Edition, Prentice Hall of India/MIT Press.
4. Aho, A. V., Hopcroft, J. E., and Ullman, J. D., Data Structures and Algorithms, Addison-Wesley.
5. Levitin, A., Introduction to the Design and Analysis of Algorithms, 2nd Edition, Pearson.

Online/E-Resourses:

Malaviya National Institute of Technology Jaipur
Department of Artificial intelligence and data engineering

Course Details
Introduction to Artificial Intelligence | 26AIT151 | 3 0 0 0

PREREQUISITE:
NIL

COURSE OBJECTIVE(S):
The course aims to introduce the fundamental concepts, history, and foundations of Artificial Intelligence. It develops problem-solving skills using search, adversarial search, constraint satisfaction, knowledge representation, logic, and planning. It also provides an understanding of probabilistic reasoning, utility theory, temporal models, and basic machine learning concepts.

COURSE OUTCOMES:
CO1 : Explain the foundations, history, agents, and environments of Artificial Intelligence.
CO2 : Apply search techniques, adversarial search, and constraint satisfaction methods to solve AI problems.
CO3 : Use logic, inference, knowledge representation, and planning for intelligent reasoning
CO4 : Apply probability, temporal models, Hidden Markov Models, and utility theory for reasoning under uncertainty.
CO5 : Analyze basic machine learning concepts, hypothesis evaluation, and learning from examples.

SYLLABUS:

UNIT I (8 Lectures): What is AI, Foundation of AI and its history, Agents and Environment

UNIT II (8 Lectures): Solving problem by searching, Beyond classical search, Adversarial search, Constraint satisfaction problems

UNIT III (9 Lectures): Quantifying uncertainty, Probabilistic reasoning, Probabilistic reasoning over time, Inference in temporal models, Hidden Markov models, The basis of utility theory, Utility functions, Multi-attribute utility functions

UNIT IV (9 Lectures): Quantifying uncertainty, Probabilistic reasoning, Probabilistic reasoning over time, Inference in temporal models, Hidden Markov models, The basis of utility theory, Utility functions, Multi-attribute utility functions

UNIT V (8 Lectures): Learning from examples, Evaluating and choosing the best hypothesis, The theory of learning, Knowledge in learning

EVALUATION
Final grades will be weighted average of performance in

Course Work	20%
Midterm Exam	30%
Endterm Exam	50%

depending on the institute's academic policy.

Textbook(s):

Reference book(s):

1. Artificial Intelligence: A Modern Approach, 3rd Edition, Stuart Russell and Peter Norvig
2. Elaine Rich, Kevin Knight, and Shivashankar B. Nair, Artificial Intelligence, McGraw Hill, 3rd Edition, 2009.
3. The Elements of Statistical Learning, Trevor Hastie and Robert Tibshirani, 2nd Edition, Springer.

Online/E-Resourses:

Malaviya National Institute of Technology Jaipur
Department of Artificial intelligence and data engineering

Course Details
Mathematics for AIDE | 26AIT152 | 3 0 0 0

PREREQUISITE:
NIL

COURSE OBJECTIVE(S):
The course aims to build a strong mathematical foundation required for Artificial Intelligence and Data Engineering. It introduces linear algebra, matrix decompositions, vector calculus, probability, and optimization techniques. Students will learn to connect these mathematical tools with AI applications such as regression, PCA, parameter estimation, GMMs, and SVMs.

COURSE OUTCOMES:

CO1 : Apply linear algebra concepts to solve AI and data engineering problems.

CO2 : Analyze matrices using eigenvalues, eigenvectors, SVD, and low-rank approximation.

CO3 : Apply vector and matrix calculus for gradients, optimization, and backpropagation.

CO4 : Apply probability, statistics, and optimization techniques to model uncertainty and solve AI problems.

CO5 : Apply mathematical tools to AI applications such as regression, PCA, GMM, and SVM.

SYLLABUS:

UNIT I (12 Lectures): Systems of linear equations, matrices and matrix operations, vector spaces, linear independence, basis and rank, linear mappings and affine spaces, norms and inner products, lengths and distances, angles and orthogonality, orthonormal bases, orthogonal complements, and projections.

UNIT II (12 Lectures): Determinant and trace, eigenvalues and eigenvectors, eigendecomposition and diagonalization, singular value decomposition, low-rank matrix approximation, differentiation of univariate and multivariate functions, partial derivatives and gradients, gradients of vector-valued functions and matrices, useful matrix-calculus identities, backpropagation and automatic differentiation, higher-order derivatives, and multivariate Taylor series.

UNIT III (9 Lectures): Probability spaces, discrete and continuous random variables, joint, marginal and conditional probability, sum rule, product rule and Bayes theorem, independence and conditional independence, expectation, variance and covariance, Gaussian distribution, exponentialfamily distributions, and change of variables.

UNIT IV (9 Lectures): Optimization using gradient descent, constrained optimization and Lagrange multipliers, fundamentals of convex optimization, empirical risk minimization and parameter estimation, mathematical formulation of linear regression, dimensionality reduction using principal component analysis, and brief mathematical connections to Gaussian mixture models and support vector machines.

EVALUATION
Final grades will be weighted average of performance in

Course Work	20%
Midterm Exam	30%

Endterm Exam	50%

depending on the institute's academic policy.

Textbook(s):

Reference book(s):

1. Marc Peter Deisenroth, A. Aldo Faisal and Cheng Soon Ong, Mathematics for Machine Learning, Cambridge University Press, 2020.
2. Gilbert Strang, Introduction to Linear Algebra, Wellesley-Cambridge Press.
3. Stephen Boyd and Lieven Vandenberghe, Convex Optimization, Cambridge University Press.
4. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer.

Online/E-Resourses: