Teaching Plan (January 2026 - June 2026)

Department of Information Technology
LCB College, Maligaon, Guwahati

Course: Bachelor of Science (Information Technology)
(As per GU-FYUGP)
Programme: NEP- B.Sc. IT
Semester: 4

Paper Title: Design and Analysis of Algorithms
Course Level: 200-299
Course Type: Compulsory
Paper Code: TBA
Theory Credit: 4
Practical Credit: NIL

Instructor: Mr. Manjit Kumar Nath

Total Hours: 60 (Theory-60 + Practical-0)

UNIT 1 — Introduction (6 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
1	Analysis of algorithms: definition, importance	Lecture + examples	Quick quiz
2	Worst-case & average-case analysis	Chalkboard worked examples	Problem exercise
3	Time complexity: counting operations	Tutorial + in-class problems	Short test Q
4	Space complexity	Lecture + examples	Quiz
5	Asymptotic notations: O, Θ, Ω — formal definitions	Proof walkthroughs	Homework proof problem
6	Proving correctness of algorithms (loop invariants, induction)	Tutorial & examples	Unit mini-assignment

UNIT 2 — Algorithm Design Techniques (10 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
7	Iterative techniques; approach & examples	Lecture + practice	Quiz
8–9	Divide and conquer: recurrence relations, Master theorem	Worked examples + problems	Assignment: solve recurrences
10–11	Dynamic programming: memoization vs tabulation; examples (LCS)	Step-by-step DP derivation	Class exercise (LCS)
12–13	Greedy algorithms: strategy, correctness proofs	Examples (activity selection, knapsack approx)	Problem set
14	Applications: sorting & searching via techniques recap	Tutorial	Short test
15	Matrix multiplication (divide & conquer vs DP overview)	Lecture	Quiz
16	Knapsack problem (DP & approximate greedy)	Problem solving	Assignment due

UNIT 3 — Sorting & Searching Techniques (20 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
17	Bubble & Insertion Sort: algorithm, complexity	Demo + trace-through examples	In-class trace exercise
18–19	Merge Sort: divide & conquer, correctness, complexity	Worked examples + recurrence solving	Problem set
20–21	Heap Sort: heapify, build-heap, delete-max	Code walkthrough / heap diagrams	Assignment (heap operations)
22–24	Quick Sort: partition schemes, average/worst case, tail recursion	Code & pivot strategy discussion	Quiz + problem
25–27	Sorting in linear time: Counting, Radix, Bucket sorts — when applicable	Examples & stability discussion	Exercise
28	Searching techniques: binary search variants, interpolation search	Lecture + problems	Short test
29–30	Medians & order statistics: selection algorithms, linear-time select	Algorithm derivation + proof	Assignment
31–33	Complexity analysis comparisons; stability, in-place considerations	Comparative discussion	Quiz
34–36	Practical considerations: library sorts, hybrid sorts (intro to Timsort)	Case study	Class exercise
37	Unit review & consolidated problem session	Revision	Unit quiz

(Note: UNIT 3 uses 20 lectures — detailed walkthroughs + practice problems to build algorithmic intuition.)

UNIT 4 — Balanced Trees (9 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
38	Need for tree balancing; AVL vs Red-Black trees overview	Lecture + examples	Quiz
39	Height properties of Red-Black trees	Proof sketch & examples	Problem exercise
40	Rotations: left rotation (mechanics & effect)	Diagram + code snippet	In-class tracing
41	Rotations: right rotation	Demo + practice	Quiz

Lecture No.	Topics Covered	Mode of Teaching	Assessment
42–43	Insertion in Red-Black trees (cases)	Stepwise walkthrough	Assignment (trace insert)
44–45	Deletion in Red-Black trees (cases)	Worked examples & practice	Short test
46	Unit review & problem session	Revision	Unit quiz

UNIT 5 — Graph Algorithms (9 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
47	Graph representations: adjacency matrix & adjacency list	Lecture + diagrams	Quiz
48	Operations cost: degree, indegree/outdegree computations	Complexity analysis examples	Exercise
49–50	Breadth-First Search (BFS): algorithm, queue usage, applications (shortest paths in unweighted graphs)	Demo + walkthrough	Assignment (BFS trace)
51–52	Depth-First Search (DFS): recursion, discovery/finish times, applications (topological sort, SCC intro)	Code walkthrough	Quiz / problem
53	Applications of traversals: connectivity, bipartiteness testing	Examples	Short test
54	Unit review & combined traversal problems	Problem solving	Unit quiz

---

UNIT 6 — String Processing (6 Lectures)

Lecture No.	Topics Covered	Mode of Teaching	Assessment
55	String matching problem: naive approach & complexity	Lecture + examples	Quiz
56–57	Knuth-Morris-Pratt (KMP) algorithm: prefix function / lps array construction	Step-by-step derivation + examples	Assignment (compute lps)
58	KMP search phase & complexity proof	Demo with examples	In-class exercise
59	Other remarks: practicalities and brief intro to advanced string algorithms	Lecture	Short test
60	Course wrap-up, revision & mock test	Revision	Mock test / feedback

---