NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	BTech CSE, Integrated BTech (CSE)-MBA
Course Code:	XXXX
Course Title:	Design and Analysis of Algorithms
Course Type:	Core
Year of Introduction:	2024-25

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	2	-	-	-	4

Course Learning Outcomes (CLO):

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

1. explain the notion of algorithmic complexity and the logic of fundamental algorithms (BL2)
2. identify suitable data structures to solve a problem effectively andefficiently (BL3)
3. apply an optimal solution approach for complex problems (BL4)
4. formulate an appropriate algorithm for real-life problems. (BL6)

Unit	Contents	Teaching Hours (Total 45)
Unit-I	**Elementary Algorithmic:** Efficiency of Algorithms, average and worst-case analysis, Elementary Operation **Analysis Techniques:** Empirical, mathematical, Asymptotic analysis and related unconditional and conditional notations.	07
Unit-II	**Analysis of Algorithms:** Analyzing control structures: sequencing, "For" loops, Recursive calls, "While" and "repeat" loops, Amortized analysis **Solving Recurrences:** Intelligent guesswork, Homogeneous recurrences, non-homogeneousRecurrences, Change of variable, Range transformations, Master Theorem, Recurrence Tree	07
Unit-III	**Advanced Data Structures:** Red-black tree, Interval tree, Binomial heaps, Fibonacci Heap, disjoint set structures. **Divide-and-Conquer:** Multiplying large integers, merge sort, quick sort, median-of-median approach, Strassen's matrix multiplication, exponentiation.	09
Unit-IV	**Dynamic Programming:** The principle of optimality, 0/1 Knapsack Problem, Assembly line Scheduling Problem, Matrix Chain Multiplication, Longest Common Subsequence, All-pairs shortest path: Floyd-Warshal's algorithm.	07
Unit-V	**Greedy Algorithms:** Activity Selection Problem, Fractional Knapsack problem, Huffman Coding, Graphs: Minimum spanning trees-Kruskal's algorithm, Prim's algorithm, Single Source Shortest paths: Bellman-Ford algorithm, Dijkstra's algorithm.	08
Unit- VI	**Branch and Bound, Backtracking**: Travelling salesman problem, n-queen problem, sum of subset problem, graph coloring problem. Theory of NP-Completeness, **Randomized and Approximation Algorithms:** Design of some classical problems.	07

Self-Study:
The self-study contents will be declared at the commencement of the semester. Around 10% of the questions will be asked from self-study contents.

Suggested Readings/ References:

1. Charles E. Leiserson, Thomas H. Cormen, Ronald L. Rivest, Clifford Stein, Introductionto Algorithms, PHI
2. Gilles Brassard & Paul Bratley, Fundamentals of Algorithmic, PHI.
3. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekharan, Fundamentals of Computer Algorithms, Galgotia.
4. Robert Sedgewick and Kevin Wayne - Algorithms, Addison Wesley
5. Rod Stephens - Essential Algorithms: A Practice Approach to Computer Algorithms Using Python and C#, Wiley

Laboratory Work:
Laboratory work will be based on the above syllabus with a minimum of 10 experiments to be incorporated. The students in a suitable group size will design and perform one experiment as a part of Laboratory work.

Sr. No.	List of Experiments/Exercises	Hours
1	Various applications of Arrays and Matrices	02
2	Working with Linked List	02
3	Searching (binary, ternary, and hash search)	02
4	Different applications of fundamental Sorting Algorithms	02
5	Use of Recursion	04
6	Divide and conquer applications and complexity computations	04
7	Applications of Greedy algorithms	04
8	Applications of Dynamic Programming	04
9	Working with tree algorithms	02
10	Working with Graph Algorithms	04