EC311: Algorithms Design and Analysis

Details of course:-

Course Title	Course Structure			Pre-Requisite: Nil
	L	T	P	
Algorithms Design and Analysis	3	1	0	NIL

Course Objective: Upon completion of this course, students will be able to do the following:

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations

Course Outcomes:

CO1: Analyze the correctness, efficiency and asymptotic runtime complexity of algorithms including formulating recurrence relations.

CO2: Understand and analyse different sorting algorithms.

CO3: Understand and design algorithms using greedy strategy, divide and conquer approach, dynamic programming, demonstrate a familiarity with major algorithms and data structures.

CO4: Able to Describe the classes P, NP, and NP Complete and be able to prove that a certain problem is NP-Complete.

S. No.	Contents	Contact Hours
Unit 1	Introduction: Concept of algorithmic efficiency, run time analysis of algorithms, Asymptotic Notations. Growth of Functions, Master's Theorem	06
Unit 2	Searching and Sorting: Structure of divide-and-conquer algorithms; examples: binary search, quick sort, Stassen Multiplication; merge sort, heap sort and Analysis of divide and conquer run time recurrence relations.	06
Unit 3	Greedy Method: Overview of the greedy paradigm examples of exact optimization solution: minimum cost spanning tree, approximate solutions: Knapsack problem, Kruskal's algorithm and Prim's algorithm for finding Minimum cost Spanning Trees, Dijkstra's and Bellman Fort Algorithm for finding Single source shortest paths.	10
Unit 4	Dynamic programming: Principle of dynamic programming. Applications: Floyd-Warshall algorithm for all pair shortest paths. Matrix multiplication, Traveling salesman Problem, longest Common sequence, Back tracking: Overview, 8-queen problem, and Knapsack problem.	06
Unit 5	Branch and bound: LC searching Bounding, FIFO branch and bound, LC branch and bound application: 0/1 Knapsack problem.	06
Unit 6	Computational Complexity: Complexity measures, Polynomial Vsnon-polynomial time complexity; NP-hard and NP-complete classes, examples: Circuit Satisfiablity, Vertex cover, Subset Sum problem, Randomized Algorithms, String Matching, NP-Hard and NP-Completeness, Approximation Algorithms, Sorting Network, Matrix Operations, Polynomials and FFT, Number Theoretic Algorithms.	08
	Total	42

Books: -

S. No	Name of Books/Authors/Publisher
1	Introduction toAlgorithms/T .H .Cormen, C .E .Leiserson, R .L .Rivest/ PHI/Fourth edition/2022
2	Fundamentals of Computer Algorithms/E. Horowitz, S. Sahni, and S. ajsekaran/Galgotia Publication/Second edition/2019
3	Computer Algorithms/Sara Basse, A. V. Gelder/Addison Wesley/Third Edition/2002