Paper Code: ARO 375	L	T/P	Credits
Subject: Analysis and Design of Algorithm	3	0	3

Marking Scheme:
Teachers Continuous Evaluation: As per university examination norms from time to time.
End Term Theory Examination: As per university examination norms from time to time.

INSTRUCTIONS TO PAPER SETTERS: Maximum Marks : AS per University norms

- There should be 9 questions in the end term examination question paper
- Question No. 1 should be compulsory and cover the entire syllabus. This question should have objective or short answer type questions.
- Apart from Question No. 1, the rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions. However, students may be asked to attempt only 1 question from each unit.
- The questions are to be framed keeping in view the learning outcomes of course/paper. The standard/ level of the questions to be asked should be at the level of the prescribed textbooks.
- The requirement of (scientific) calculators/ log-tables/ data-tables may be specified if required

Course Outcomes [Bloom's Knowledge Level (KL)]:
CO1: Ability of students to understand and evaluate the concepts complexity of algorithm and types of sorting algorithm **[K1, K5].**
CO2: Ability of students to understand and apply the concept of Dynamic Programming **[K2, K3].**
CO3: Ability of students to analyze the Greedy Algorithms **[K4].**
CO4: Ability of students to understand the concept of NP-Complete Problem **[K2].**

CO/PO	PO01	PO02	PO03	PO04	PO05	PO06	PO07	PO08	PO09	PO10	PO11	PO12
CO1	3	3	3	3	2	-	-	-	-	1	1	2
CO2	3	3	3	3	2	-	-	-	-	1	1	2
CO3	3	3	3	3	2	-	-	-	-	1	1	3
CO4	3	3	3	3	2	-	-	-	-	1	1	3

Course Content	**No of lectures**
Unit I Asymptotic notations for time and space complexity, Big-Oh notation, Θ notation, Ω notation, the little-oh notation, the little-omega notation, Recurrence relations: iteration method, recursion tree method, substitution method, master method, Data Structures for Disjoint Sets,. Complexity analysis, Insertion sort, Merge Sort, Quick sort. Strassen's algorithm for Matrix Multiplications.	[10]

Unit II Ingredients of Dynamic Programming, emphasis on optimal substructure , overlapping substructures, memorization. Matrix Chain Multiplication, Longest common subsequence and optimal binary search trees problems, 0-1 knapsack problem, Binomial coefficient computation through dynamic programming. Floyd Warshall algorithm.	[10]
Unit III Greedy Algorithms: Elements of Greedy strategy, overview of local and global optima, matroid, Activity selection problem, Fractional Knapsack problem, Huffman Codes, A task scheduling problem. Minimum Spanning Trees: Kruskal's and Prim's Algorithm, Single source shortest path: Dijkstra and Bellman Ford Algorithm.	[10]
Unit IV The naïve String Matching algorithm, The Rabin-Karp Algorithm, String Matching with finite automata, The Knuth-Morris Pratt algorithm.	[8]

Text Books:

[T1] Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to algorithms. MIT press.*

[T2] Kleinberg, J., & Tardos, E. (2006). *Algorithm design*. Pearson Education India.

Reference Books:

[R1] Baase, S. (2009). *Computer algorithms: introduction to design and analysis*. Pearson Education India.