Course Title	CS854 Advanced Algorithm Analysis
Credit Hours	3+0
Course Description:	This course concentrates on designing an algorithm for computational problem solving by developing in-depth knowledge of the problem domain, available data structures, algorithmic design techniques, formal analysis techniques and related underlying mathematical theory.
Learning Outcomes:	On successful completion of this course students will be able to: 1. Develop good understanding on a wide range of advanced algorithmic problems, their mathematical concepts, relations, variants, and complexities. 2. Should develop sound theoretical understanding of advanced algorithms along with practical problem-solving skills needed for application of appropriate algorithm to a practical situation 3. Design complex and intricate algorithms.
Textbooks:	Introduction to Algorithms, By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 3rd edition, Published by MIT Press.
Reference Books:	▪ Approximation Algorithms, By Vijay V. Vazirani, Springer. ▪ Algorithms and Theory of Computation Handbook, By Mikhail J. Atallah, CRC Press.
Course Contents:	▪ Review of Sorting and Searching Algorithms ▪ Algorithm Analysis: Big-O, Small-o, Big-theta • Recurrences • The substitution method • The recursion-tree method • The master method ▪ Probabilistic Analysis and Random Algorithms • Indicator random variables • Randomized algorithms ▪ Dynamic Programming • Assembly-line scheduling • Matrix-chain multiplication • Elements of dynamic programming ▪ Greedy Algorithms • Activity selection problem • Elements in greedy strategy • Huffman codes ▪ Amortized Analysis • Aggregate analysis • The accounting method • The potential method ▪ NP completeness • Polynomial time and NP problems, completeness proof, reducibility. • Non-Computable function

	• Halting Problem • implications of non-computability ▪ Approximation Algorithms • The vertex cover problem • The traveling salesman problem • The set covering problem ▪ Pattern Matching • Naïve string search • Finite-state-automaton-based search • Index method • Stubs ▪ Numerical approximations • Direct and iterative methods • Linear Programing • Semi definite Programing ▪ Optimization • backtracking • branch-and-bound ▪ Graphs • Minimum spanning Tress • Shortest Path ▪ Network Flows • Minimum Flow • Maximum flow ▪ Multithreaded Algorithms • Dynamic multithreading model • Metrics of work, span, and parallelism • Matrices multiplication with multithreading