EVEN 2022 - Lecture Plan

15B11CI411 – Algorithms and Problem Solving

Course Coordinator – Dr. Tribhuwan K. Tewari (J62) / Dr. Pulkit Mehndiratta (J128)

Lecture #	Title of the Module	Topics to be Discussed
Lecture 1	Introduction	General discussion about time and space requirements in an algorithm. Asymptotic notations (Big O, Big Omega, etc.)
Lecture 2		Asymptotic analysis using various interactive algorithms, e.g. linear search, finding maximum and minimum elements in an array, etc.
Lecture 3		Finding recurrences for various recursive algorithms, e.g. binary search, merge sort, quick sort, median search, heap sort, etc.
Lecture 4		Solving recurrences using substitution method
Lecture 5		Solving recurrences using tree substitution method
Lecture 6		Solving recurrences using Master’s Theorem method
Lecture 7	Search Trees	Query efficient data structures, segment tree and interval tree.
Lecture 8		Various operations, e.g. insert, range search, etc. on segment tree
Lecture 9		Various operations, e.g. insert, range search, etc. on interval tree
Lecture 10	Design Technique: Divide and Conquer	Fundamentals of Divide and Conquer (D&C) approach. Review of D&C based searching
Lecture 11		Divide and Conquer sorting techniques (Merge Sort and Quick Sort)
Lecture 12		D&C based Strassen’s matrix multiplication
Lecture 13		finding closest pair in 1D and 2D search spaces

Lecture 14	Design Technique: Backtracking Algorithms	Depth first and breadth first searches in graph. Review of the backtracking algorithms using N queen and rat in a maze problem. Backtrack based solution approach for the graph colouring problem
Lecture 15		Backtracking based strategy for finding Hamiltonian Cycle in un-weighted graph. Solving traveling salesman problem using backtracking
Lecture 16		Finding maximum flow in the flow network using DFS and BFS. Discussion of the residual flow network
Lecture 17		Computation of the maximum flow in the given flow network using Ford Fulkerson and Edmond Karp algorithms
Lecture 18	Design Technique: Greedy Algorithms	Introduction to the greedy design technique, effectiveness of the objective function, finding minimum spanning tree (MST) using Kruskal's algorithm, efficient data structures for implementation of Kruskal's algorithm
Lecture 19		Prim's algorithm for finding the MST. Effectiveness of different data structures for implementation of the Prim's algorithm
Lecture 20		Finding single source shortest path using greedy design technique based Dijkstra's algorithm, objective functions for solving Knapsack (0/1 and fractional) and coinage problems
Lecture 21		Discussion of the various objective functions (first fit, best fit, etc.) for solving packing problems in 1D (Strip packing), 2D & 3D (bin packing and container loading). Greedy algorithm (shortest job first, etc.) for solving scheduling problems
Lecture 22		Solving the problem of graph colouring using greedy design technique. Discussion of the applications of the graph colouring in scheduling problems, e.g. time table scheduling
Lecture 23		Greedy design technique based schemes (Huffman and Shannon-Fano) for text compression
Lecture 24	Dynamic Programming	Fundamentals of Dynamic Programming, e.g. sub-problem identification, overlapping sub-problems/states, memorization scheme, and

		recurrence. Developing the recurrence for solving the 0/1 Knapsack problem
Lecture 25		Solving 0/1 Knapsack and Coinage problem using Dynamic Programming
Lecture 26		Discussion of the all source all destination shortest path using dynamic programming
Lecture 27		Finding longest common subsequence in given two strings using dynamic programming
Lecture 28		Developing the dynamic programming based solution strategy for Matrix Chain Multiplication and String Editing problem
Lecture 29		Finding longest increasing sequence using dynamic programming. Discussion of the problems based on LIS
Lecture 30	String Algorithms	Real life applications of String matching, Brute force approach for finding patterns in given text
Lecture 31		Pattern matching using finite automata. Finding multiple patterns in a given text using finite automata
Lecture 32		Approach of Knuth Morris Pratt for pattern matching
Lecture 33		Hashing based approach (Rabin Karp) for pattern matching
Lecture 34		Data structures (Tries and compressed tries) for string
Lecture 35		Suffix tree and Suffix array data structures and related operation for string handling
Lecture 36		String handling using Suffix tree and Suffix array
Lecture 37	Problem Spaces and Problem solving by search	Introduction to problem solving, search space, discussion of various puzzles e.g. 8-puzzle, tic-tac-toe, etc. as examples of problem solving
Lecture 38		Discussion of uninformed search and informed search, relevance of heuristics for implementing the informed search, informed search using Hill climbing, solving puzzles using hill climbing
Lecture 39		Informed search using Best first search. Analysis of the performance of the hill climbing and best first search techniques of informed search

Lecture 40		Discussion of A* algorithm. Finding shortest path using A* algorithm
Lecture 41	Tractable and Non-Tractable Problems	Discussion of the NP completeness and NP hard problems, listing of various NP hard problems, viz. vertex cover, graph color, Hamiltonian cycles, etc.
Lecture 42		Solution approaches for NP hard problems