DAA Tutorial-4
Greedy Approach

Note: For each problem, you are required to implement the solution in C++, present the complete code, and demonstrate its execution to the evaluator.

Problem 1: Fractional Knapsack Problem

Problem Definition: Given n items with values v_i and weights w_i and a knapsack with capacity W, the goal is to maximize profit by selecting items. Unlike 0/1 Knapsack, items can be broken into fractions. The greedy approach sorts items by value-to-weight ratio.

Example: Items = $\{(v = 60, w = 10), (v = 100, w = 20), (v = 120, w = 30)\}$, $W = 50$ Take items 2 and 3 fully, and half of item 1. Optimal Profit = 240.

Problem 2: Job Sequencing with Deadlines

Problem Definition: You are given n jobs, each with a deadline d_i and profit p_i. Each job takes one unit of time to finish, and a job can only be scheduled if it is finished before its deadline. The goal is to maximize the total profit.

Example: Jobs = $\{(d = 2, p = 100), (d = 1, p = 19), (d = 2, p = 27), (d = 1, p = 25), (d = 3, p = 15)\}$ Optimal Schedule = Jobs 1 and 3 (Profit = 127).

Problem 3: Huffman Coding (Greedy File Compression)

Problem Definition: Given character frequencies, construct an optimal prefix code using Huffman's algorithm, minimizing the total cost of encoding.

Example: Characters = $\{a : 5, b : 9, c : 12, d : 13, e : 16, f : 45\}$ Huffman cost = 224.

Problem 4: Cell Tower Placement

Problem Definition: Along a road with scattered houses, place the minimum number of cell towers so that every house is within 4 miles of a tower.

Example: Houses = $[1, 2, 6, 9, 11]$ Towers placed at $[2, 9]$ cover all houses.

Problem 5: Minimum Refuelling Stops

Problem Definition: A car can travel n km on a full tank. Petrol pumps are placed along the route with known distances. The goal is to minimize the number of stops needed to reach the destination.

Example: Tank = 10 km, Distance = 25 km, Petrol Pumps at $[4, 7, 12, 18, 22]$ Minimum stops = 2 (stop at 7 and 18).

Problem 6: License Purchasing Problem

Problem Definition: You must buy n software licenses, one per month. Each license initially costs Rs. 1000 but its cost increases every month at an exponential rate $r_j (> 1)$. Determine the order in which the licenses should be purchased so that the total cost is minimized.

Example: Rates = $[2, 3, 1.5]$ Find the optimal order of purchase that results in the minimum total expenditure.