# UNIT-IV
# DYNAMIC PROGRAMMING

1) Explain the methodology of dynamic programming. Mention the applications of Dynamic programming. **(BTL - II, V)**
2) What is All-Pair shortest path problem (APSP)? Discuss the Floyd's APSP algorithm. **(BTL - I, VI )**
3) Discuss about All-Pairs Shortest Paths. **(BTL - VI )**
4) Find the all pairs shortest path solution for the graph represented below adjacency matrix:

$$\begin{bmatrix} \infty & 6 & 5 & 4 \\ 3 & \infty & 2 & 6 \\ 18 & 6 & \infty & 7 \\ 8 & 12 & 10 & \infty \end{bmatrix}$$ **(BTL - VI)**

5) Apply dynamic programming to obtain optimal binary search tree for the identifier set $(a_1, a_2, a_3, a_4) = (cin, for, \text{int}, While)$ with $(p_1, p_2, p_3, p_4) = (1,4,2,1)$, $(q_0, q_1, q_2, q_3, q_4) = (4,2,4,1,1)$ and also write algorithm for its construction. **(BTL - III, I)**
6) Draw an Optimal Binary Search Tree for $n = 4$ identifier set $(a_1, a_2, a_3, a_4) = (do, if, read, while)$ with $(p_1, p_2, p_3, p_4) = (3,3,1,1)$, $(q_0, q_1, q_2, q_3, q_4) = (2,3,1,1,1)$ **(BTL – VI)**
7) Describe the Dynamic 0/1 knapsack problem. Find an optimal solution for the dynamic programming 0/1 knapsack instance for $n = 3, m = 6$, profits are $(p_1, p_2, p_3) = (1,2,5)$, weights are $(w_1, w_2, w_3) = (2,3,4)$ **(BTL - II )**
8) Explain 0/1 knapsack problem briefly. **(BTL - II, V)**
9) Describe about reliability design with an example. **(BTL - II )**
10) What is the need of Reliability Design in Dynamic Programming? Discuss. **(BTL - I, VI)**
11) What is principle of optimality? Explain travelling sales person problem. **(BTL – I,II, V)**
12) Construct an optimal travelling sales person tour using Dynamic Programming

$$\begin{bmatrix} 0 & 10 & 9 & 3 \\ 5 & 0 & 6 & 2 \\ 9 & 6 & 0 & 7 \\ 7 & 3 & 5 & 0 \end{bmatrix}$$ **(BTL - III, IV)**

13) Discuss the time and space complexity of Dynamic programming travelling sales person algorithm. **(BTL - VI )**
14) Explain about travelling Sales Person problem with an example. **(BTL - II, V)**
15) Explain how matrix-chain multiplication problem can be solved using dynamic programming with suitable example. **(BTL - II, V)**