1. (a) What is the time complexity of Huffman-Coding algorithm for finding the optimal code for an alphabet with \( n \) symbols? Give a brief explanation.

(b) Which of the following design strategies does the algorithm use?
   i. Divide-and-Conquer
   ii. Greedy
   iii. Dynamic Programming
   iv. None of the above

(c) Obtain a Huffman-Code for the following alphabet with the given probabilities. Show how the code is obtained. (Show the corresponding code tree.)
Given an array of \( n \) elements with arbitrary random values. We want to obtain the smallest \( \sqrt{n} \) elements in sorted order. Outline how each of the following methods may be adapted to solve this problem (if appropriate), and provide a very brief explanation. If the method is not applicable, state “NA” and provide a one-sentence explanation. Analyze the worst-case time complexity for each of the appropriate cases. (State the time complexity for each major step in your outline.)

(a) Insertion Sort

(b) An adaptation of Bubble-Sort.

(c) Use a HEAP.

(d) Use the linear-time SELECTION algorithm (which used the median-of-medians for pivot).

(e) Which of the above methods is best-suited and most-efficient for this problem? Explain.
3. Given an undirected graph with \( n \) vertices and \( e \) edges.

(a) Outline an algorithm to find the Connected-Components (CC) of the graph in the form of an array \( A \) such that \( A[i] = A[j] \) if and only if vertices \( i \) and \( j \) are in the same component.

(b) Analyze the time complexity for the case when the graph is represented by its adjacency lists.

(c) Analyze the time complexity for the case when the graph is represented by its adjacency matrix.
4. Floyd’s all-pairs-shortest-paths algorithm for a weighted directed graph produces two matrices $A$ and $P$, where $A[i, j]$ is the cost of the shortest $(i, j)$ path and $P[i, j]$ is the next vertex after vertex $i$ on the shortest $(i, j)$ path.

(a) What is the time complexity of Floyd’s algorithm?
(b) Illustrate the algorithm for the following graph. (Show matrices $A$ and $P$ after each iteration.)
5. Consider a connected, undirected weighted graph, represented by its adjacency lists.

(a) Find a minimum-cost-spanning tree (MST) for the following graph using Kruskal’s algorithm. Show the result after each iteration.

(b) Find an MST for the same graph using Baruvka’s algorithm. Show the result after each round.
(c) What is the time complexity of Baruvka’s algorithm, assuming that the adjacency list for each vertex is already sorted by the edge weights?

Provide a brief analysis of the time complexity.