Problem 1 (20pt)

Betweenness centrality is a metric of importance of a node in a graph. For a node v, the betweenness centrality BC(v) is the number of shortest paths in the network to which node v belongs. Formally,

\[ BC(v) = \sum_{s,t \in V} \sigma_{s,t}(v) \]

where \( \sigma_{s,t}(v) = 1 \) if v belongs to the shortest path between s and t, 0 otherwise.

Write the pseudo code of an algorithm that takes as input the predecessor matrix \( \Pi^{(n)} \) returned by the Floyd-Warshall algorithm and a node v, and returns the betweenness centrality BC(v).

You may assume there is a single shortest path between any two pair of nodes.
Problem 2 (10pt)
Describe the Minimum Spanning Tree (MST) problem and the Krusckal's algorithm for such problem. A complete answer should include:
(i) Definition of the MST problem (5pt)
(ii) Description of the Krusckal's algorithm (5pt)

Problem 3 (10pt)
Describe the optimality principle and its use in the Floyd-Warshall algorithm.
Problem 4 (10pt)
Describe the meaning of the element \([i,j]\) of the matrix \(D^{(k)}\) in the Floyd-Warshall algorithm, provide the recursive relation to calculate such element and motivate this relation.

Problem 5 (10pt)
Show a graph for which the minimum spanning tree is different from the shortest path tree from a given source. Depict both trees.
Problem 6 (10pt)
Apply Dijkstra Algorithm to the following graph. To this purpose, write the value of the attributes on each node as set by the algorithm. Clearly highlight the resulting shortest path tree. Use node 1 as the source.
Problem 7 (15pt)
Show the execution of the Floyd-Warshall algorithm on the following graph. To this purpose, fill the values of the provided tables. Tables on the right represent the D tables, those on the left the Π tables.
Problem 8 (15pt)
Show the execution of the Ford-Fulkerson algorithm to calculate the Maximum Flow of the following flow network. Show for each iteration of the algorithm the current flow assignment and the corresponding residual network.