FACULTY OF ENGINEERING SCIENCES AND TECHNOLOGY

Department: **Computer Science** Program: **BS(CS)**

Discrete Structure

Announced date: 30 – Dec – 2025 **Due Date: 20 – Jan – 2026** Max Marks: 10

Complex Computing Problem (CCP)		
Mapped CLO	**SDG**	**Complex Problem Solving Mapped**
CLO 1 – 3	4	WP1 (Depth of knowledge required) WP2 (Range of conflicting requirements required) WP3 (Depth of analysis required)

Title: Optimizing Emergency Response Times in Urban Transport Networks Using Dijkstra's Algorithm:

Problem Statement:

A city government wants to improve its emergency response system by analyzing the road network between hospitals and high-risk zones (e.g., schools, markets, factories). Each road has a specific travel time affected by traffic and distance. Your task is to model this road network as a weighted graph and apply **Dijkstra's Algorithm** to determine the shortest travel times from hospitals to various emergency points.

Tasks & Components:

1. Graph Construction:

- Model the city's road network as a weighted undirected graph**.**
- Use at least 10 nodes (e.g., locations like hospital, school, fire station, etc.).
- Each edge should have a weight representing travel time in minutes.

2. Dijkstra's Algorithm Implementation:

- Implement Dijkstra's Algorithm in Python or C++ to calculate the shortest paths from:
  - The main hospital to all other nodes.
- Output both:
  - The shortest path (sequence of nodes).
  - The minimum travel time**.**

3. Graph Visualization:

- Use a graph visualization library like NetworkX (Python) or Graphviz.
- Highlight the shortest paths on the graph.

4. Analysis & Interpretation:

- Compare shortest paths from two different hospitals.
- Discuss how the shortest paths change if one road (edge) becomes blocked or congested.
- Analyze how the system can help prioritize routes during peak hours.

Constraints:

Constraints:

- All nodes must be connected (no isolated nodes).
- Graph must contain at least 2 hospitals and 3 high-risk zones.

Expected Outcomes:

- A graph-based model of an urban transport network.
- Correct implementation of Dijkstra's algorithm.
- Visualization and analysis of optimal paths.
- Discussion on how real-time road data can further optimize the algorithm.

Learning Outcomes:

- Apply graph theory concepts to real-world transport optimization.
- Implement and explain Dijkstra's shortest path algorithm.
- Visualize graph data and interpret the results in decision-making contexts.
- Understand how to adapt algorithms to dynamic urban environments.

Report Deliverable:

A detailed report (PDF) with the following sections:

- Abstract.
- Introduction.
- Dataset description.
- Data Preprocessing.
- Model & Evaluation.
- Discussion on results.
- Conclusion.
- References.

Deliverable:

Category	Deliverables
Conceptual	Graph design with labeled nodes and edge weights
Implementation	Code for Dijkstra’s Algorithm and visualization using Network X or Graphviz
Testing	Output of shortest paths and travel times
Report	Description of network, algorithm, analysis of edge cases
Presentation	Slide deck with graph, path visuals, and recommendations

CCP Attributes mapped		
Attributes of Complex Problem Solving		**Justification**
WP	**Description**	**Discrete Structure CCP Mapping**
WP1	Depth of Knowledge	The project involves modeling real-world systems as graphs and requires a deep understanding of graph theory concepts, including weighted graphs, shortest path algorithms (Dijkstra’s Algorithm), and adjacency representations. It also demands proficiency in algorithm implementation and graph visualization tools like Network X or Graphviz, reflecting Depth of Knowledge Level 3–4 (strategic thinking and applied problem-solving).
WP2	Range of Conflicting Requirements	The task must balance algorithmic efficiency**,** path optimality, and real-world constraints such as blocked roads or variable traffic conditions. Conflicts arise in route selection (e.g., choosing fastest vs. safest paths), graph accuracy vs. abstraction, and static vs. dynamic graph updates**.** These introduce trade-offs between model simplicity and practical realism
WP3	Depth of Analysis	The analysis requires students to evaluate how network changes impact emergency response times, and compare path outputs under different scenarios**.** It pushes beyond simple implementation toward critical assessment, simulation, and interpretation**,** aligning with high-level cognitive tasks like optimizing logistics under constraints and drawing insights for public safety and infrastructure planning.