Intelligent City Navigation System Based on Graphs

Group Members:
Muhammad Musaab Ul Haq (501739)
Ahmed Hassan Raza (511263)
Muhammad Usman Amjad (516261)
Farida Mohammadi (534534)

Instructor: Dr. Ayesha Hakim

Course: CS-250 Data Structures & Algorithms

Bachelors of Science in Computer Science

National University of Sciences and Technology
November 14, 2025

Problem Statement

This project builds an intelligent navigation system that computes optimal routes by dynamically integrating terrain elevation, multi-stop routing, and multi-modal transport, providing more practical and efficient travel solutions.

Problem Description

We will design and implement a "City Navigation Graph," a multi-module software application using Python. The system will process real-world map data from OpenStreetMap to construct a weighted graph representation of a city's road network. Users will be able to specify a start and end point, along with optional intermediate stops, and receive an optimized route based on their selected mode of transport. The core of the project will be the implementation and analysis of efficient pathfinding algorithms that account for distance, topography, and user-defined constraints.

Scope and Functional Modules

The project will be developed with the following four functional modules, meeting the course requirement for at least three distinct features:

- **Core Pathfinding Engine:** Calculates the shortest path between two points based on distance.
- **Multi-Stop Route Planning:** Computes the most efficient route that connects a sequence of user-defined stops.
- **Multi-Modal and Topography-Aware Routing:**
  - Differentiates between vehicle and walking paths.
  - Integrates elevation data into edge weights, penalizing steep inclines to find topographically simpler routes.
- **Point-of-Interest (POI) Search:** Implements a feature to efficiently find and display nearby fuel stations along a calculated route.

Chosen DSA Techniques and Algorithms

To meet the project requirements, we will implement and analyze the following data structures and algorithms:

- **Graph Representation:** The core data structure will be a weighted, directed graph, implemented using adjacency lists, to model the road network. Nodes will represent intersections, and edges will represent road segments with weights corresponding to distance and elevation cost.
- **A* (A-Star) Algorithm:** This informed search algorithm will be our primary pathfinding technique. It is ideal for this problem as it uses a heuristic (Euclidean distance) to find the shortest path more efficiently than Dijkstra's. The cost function ‘g(n)‘ will be modified to include penalties for elevation changes.
- **Priority Queue (Min-Heap):** A min-heap will be implemented to serve as the priority queue for the A* algorithm's open set. This is crucial for efficiently selecting the next node to visit with the lowest cost, ensuring an optimal time complexity.
- **Spatial Hashing/Indexing:** To implement the "nearby fuel stations" feature efficiently, we will use a spatial hashing technique to store POI locations. This avoids a linear scan of all stations and allows for rapid lookups of points within a given geographical radius.