Project Title: Village Navigation and Insight System Using Graph Algorithms in C++

Overview

This project focuses on utilizing graph algorithms to develop an advanced navigation and insight system for a village. By creating a comprehensive dataset that includes 300 nodes representing different places within the village and over 700 edges acting as road segments, this project demonstrates the practical application of various algorithms and data structures in a real-world setting.

Dataset

Nodes: Each of the 300 nodes represents a specific place within the village, categorized by unique identifiers, types, and names.

Edges: Over 700 edges connect these nodes, representing the smallest units of roads. Each edge includes properties such as start and end points, length, and boolean flags indicating accessibility for different modes of transportation (walking, biking, and driving). Additionally, edges are categorized by road type (main road, highway, etc.).

Algorithms and Data Structures

The project employs several algorithms and data structures, including but not limited to:

- Graph Traversal Algorithms: Breadth-First Search (BFS) and Depth-First Search (DFS) for exploring the village layout.
- Shortest Path Algorithms: Dijkstra's algorithm for finding the minimum distance between nodes.
- Backtracking: Used in specific scenarios to explore all possible paths.
- Custom Functions: Implemented to handle various queries and provide detailed insights.

Key Functionalities

1. Connectivity and Essential Services:
   - New users can determine the connectivity of places and the distance to essential services, allowing them to judge the accessibility of different areas.

2. Shortest Path Calculation:
   - Users can find the shortest path between two places, considering different modes of transportation and specific road types.

3. Optimal Route Planning:
   - Given a list of places to visit, the system provides a sequence and route that minimizes the total distance traveled.
   - This feature is particularly useful for planning daily activities or tours.

4. Time Estimation:

- The system estimates travel time based on the mode of transportation (walking, biking, car) and provides specific routes accordingly.

- Edges in the graph include properties to indicate if a particular mode of transportation is possible, ensuring accurate time and route calculations.

Edge Class Definition

The edge class includes the following attributes:

- Start and End Points: Nodes representing the start and end of the road segment.

- Length: Distance of the road segment.

- Accessibility: Boolean flags indicating if walking, biking, or driving is possible on the segment.

- Road Type: Classification of the road (main road, highway, etc.).

Practical Application

The primary goal is to create an application that offers detailed and practical navigation and insight for a village or city. Users can access a wide range of information, such as:

- The shortest and most efficient routes to various destinations.

- Travel time estimates based on the chosen mode of transportation.

- Recommendations for the best sequence of places to visit in a day, minimizing travel distance and time.

Conclusion

This project showcases the powerful capabilities of graph algorithms and data structures in solving real-world navigation problems. By simulating a village's road network and providing essential insights, the system serves as a valuable tool for residents and visitors alike, offering a seamless and efficient way to navigate and explore the area.

This description captures the essence of your project, highlighting its components, functionalities, and practical applications in a clear and structured manner.