Professional Challenge: Smart Traffic Management System

Problem Description:

Design a system to intelligently manage traffic flow at a busy intersection. Your solution should address the following tasks:

1. Simulate the number of vehicles on each street.
2. Implement a dynamic algorithm to adjust traffic light timings based on traffic density.
3. Prioritize emergency vehicles like ambulances or fire trucks.

Requirements:

- **Programming Language:** Your choice (Python, C, Rust, or Java).
- **Inputs:**
  - Number of vehicles entering each street per second (randomly generated).
  - Identification of emergency vehicles using a specific ID (e.g., license plate).
- **Outputs:**
  - Optimized traffic light timings to minimize total delay.
  - Number of vehicles passing through the intersection during each cycle.
  - A summary report of the system's performance.

Technical Details:

1. **Traffic Simulation:**
   - Generate random data for vehicle counts.
   - Use an appropriate data structure (e.g., queue) to model vehicle movement.
2. **Traffic Light Algorithm:**
   - Develop a dynamic algorithm to adjust green and red light durations based on real-time traffic density.
   - Incorporate rules for prioritizing emergency vehicles.
3. **Performance Analysis:**
   - Calculate total vehicle delay time.
   - Provide a summary report showing the number of vehicles processed and their average wait time.

Optional Advanced Challenge:

- Use **AI or Machine Learning** to optimize the traffic management algorithm further.
- Build a simple user interface (CLI or GUI) to visualize real-time traffic at the intersection.

Objective:

This challenge is designed to test your skills in algorithm design, programming, and solving real-world problems. The goal is to create a robust and efficient system that manages traffic intelligently and minimizes delays.

https://github.com/amir13872/STMS

mahnaznamani007@gmail.com

ZARE