Traffic Light Control System

NAME – RISHI PATWA

BRANCH – CSE AIML

SECTION – C

UNIVERSITY ROLL NO – 202401100400156

KIET GROUP OF INSTITUTION

INTRODUCTION

A **Traffic Light Control System** is an automated system designed to regulate the movement of vehicles and pedestrians at road intersections using a sequence of colored lights—**red, yellow, and green**. This system ensures smooth traffic flow, minimizes congestion, and enhances road safety by assigning the right of way to different lanes at specified intervals.

In modern cities, traffic lights are managed using timers, sensors, and adaptive algorithms to dynamically control signal changes based on real-time traffic conditions. The system operates on a predefined cycle:

- **Red Light** – Stops vehicles to allow cross-traffic to move.
- **Yellow Light** – A warning that the signal is about to change.
- **Green Light** – Allows vehicles to proceed.
- A well-designed **Traffic Light Control System** can prevent accidents, improve fuel efficiency, and reduce travel delays, making road transportation more efficient and safer.

METHODOLOGY

1. **System Design & Planning** – Define traffic flow, determine signal phases, and choose hardware/software tools.
2. **Implementation Approach:**
    1. **Timer-Based Control:** Uses fixed-time intervals for red, yellow, and green lights.
    2. **Sensor-Based Control:** Adjusts timing based on real-time traffic using sensors.
    3. **AI & IoT-Based Control:** Uses AI and IoT to optimize signals dynamically.
3. **Traffic Light Simulation (Python & Tkinter)** – Develop a GUI-based system to visualize signal changes.
4. **Testing & Optimization** – Evaluate performance under different traffic conditions and fine-tune timing.
5. **Deployment & Monitoring** – Implement in real-world scenarios, monitor traffic, and optimize efficiency.

CODE & OUTPUT

```python
import time

class TrafficLight:
    def __init__(self):
        self.states = ["Red", "Green", "Yellow"]
        self.current_state = 0  # Start with Red
        self.running = True  # Control variable to stop the loop

    def change_light(self):
        start_time = time.time()  # Record start time
        while self.running:
            elapsed_time = time.time() - start_time
            if elapsed_time >= 60:  # Stop after 1 minute
                self.stop()
                print("\nTraffic Light System Stopped after 1 minute.")
                break

            state = self.states[self.current_state]
            if state == "Red":
                print("🔴 Red Light - Stop! (5 sec)")
                time.sleep(5)
            elif state == "Green":
                print("🟢 Green Light - Go! (5 sec)")
                time.sleep(5)
            elif state == "Yellow":
                print("🟡 Yellow Light - Slow down! (2 sec)")
                time.sleep(2)

            self.current_state = (self.current_state + 1) % 3  # Cycle through states
```

```python
                time.sleep(2)

            self.current_state = (self.current_state + 1) % 3  # Cycle through states

    def stop(self):
        self.running = False

# Run the Traffic Light System
if __name__ == "__main__":
    traffic_light = TrafficLight()
    traffic_light.change_light()
```

```
🔴 Red Light - Stop! (5 sec)
🟢 Green Light - Go! (5 sec)
🟡 Yellow Light - Slow down! (2 sec)
🔴 Red Light - Stop! (5 sec)
🟢 Green Light - Go! (5 sec)
🟡 Yellow Light - Slow down! (2 sec)
🔴 Red Light - Stop! (5 sec)
🟢 Green Light - Go! (5 sec)
🟡 Yellow Light - Slow down! (2 sec)
🔴 Red Light - Stop! (5 sec)
🟢 Green Light - Go! (5 sec)
🟡 Yellow Light - Slow down! (2 sec)
🔴 Red Light - Stop! (5 sec)
🟢 Green Light - Go! (5 sec)
🟡 Yellow Light - Slow down! (2 sec)
```