**Traffic Flow Simulation Project Documentation**

**1. Project Title:** Traffic Flow Simulation

**2. Objective:** To simulate vehicle movement across a simplified road network using Discrete-Event Simulation (DES) in Java, and analyze how parameters like number of cars, road capacity, and traffic light timings affect traffic flow and congestion.

**3. Tools & Technologies Used:**

* Programming Language: Java
* IDE: Visual Studio Code
* Java Collections: PriorityQueue, Lists, Maps

**4. Type of Simulation:** Discrete-Event Simulation (DES)

* Simulation time advances by jumping from one event to the next.
* Events include car movements, traffic light changes, and arrivals.

**5. System Components:**

* **Car:** Represents each vehicle; has speed, position, destination, and time details.
* **RoadSegment:** Represents roads with capacity and length; keeps track of cars on it.
* **Intersection:** Where roads connect; managed by traffic lights.
* **TrafficLight:** Manages green/yellow/red signals per direction.
* **EventQueue:** Stores and processes future events in chronological order.
* **SimulationEvent:** Abstract base for event types like CarMoveEvent and TrafficLightChangeEvent.
* **StatisticsCollector:** Records travel times, congestion, and number of completed trips.

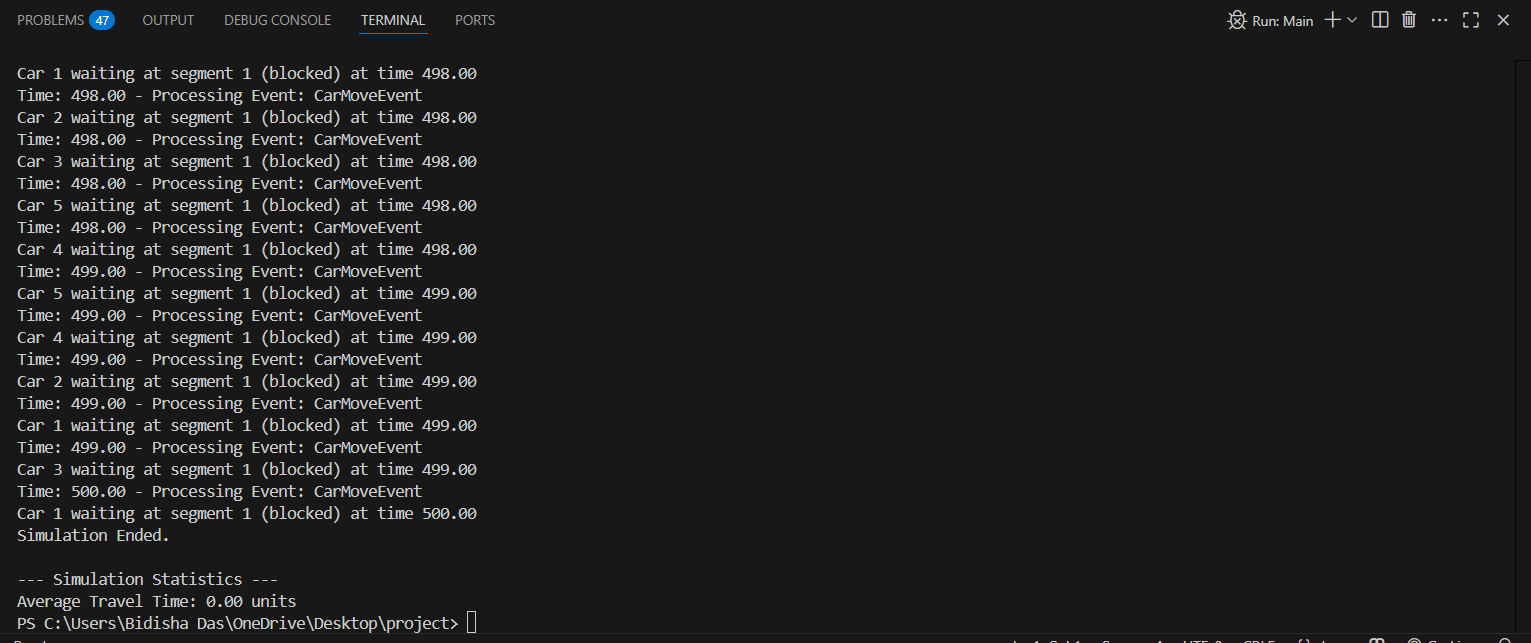
**6. High-Level Algorithm:**

* **Initialization:**
  + Build the network: intersections, road segments.
  + Create cars with random start and destination.
  + Set up traffic lights and initial event queue.
* **Simulation Loop:**
  + Process events from the queue in time order.
  + Car movement:
    - If light is green and next segment has capacity, car moves.
    - If blocked, reschedule event to try again later.
  + Traffic light change:
    - Advance phase and reschedule next light change.
    - Allow waiting cars to proceed if possible.
  + Car arrival:
    - Record travel time and remove car from system.

**7. Output:**

* Logs showing when cars move or wait.
* Summary of average travel times, number of completed trips, and congestion levels.

**8. Sample Output (Terminal):**

****

**9. Key Learning Outcomes:**

* Understanding and applying discrete-event simulation.
* Implementing object-oriented programming (OOP) principles in Java.
* Handling complex state transitions and event scheduling.

**10. Conclusion:** This simulation successfully models basic traffic behavior using Java and DES principles. It demonstrates how different traffic parameters impact overall system efficiency and helps in learning system design and time-based event control