**PSEDOCODE IMPLEMENTATION**

1. Declare necessary include statements for the classes and libraries

2. Define the main() function:

* Declare any required variables
* Create instances of the Road, Crosswalk, ParkingLot, and EmergencyService classes
* Loop until the user chooses to exit.
* Display the menu options for the traffic system.
* Read and store the user's choice in a variable
* Switch based on the user's choice:

3, Case 1: Parking

1. Prompt the user for the parking lot capacity.
2. Create a ParkingLot object with the specified capacity
3. Loop until the user chooses to exit the parking module:
4. . Display the parking options.
5. Read and store the user's choice in a variable.
6. Switch based on the user's choice:

1.1, Case "Parked":

1. Loop until the parking lot is full:
2. Prompt the user for the vehicle details
3. . Park the vehicle using the parkVehicle() method of the ParkingLot object

1.2, Case "Removed":

1, Prompt the user for the vehicle details, fee, and parking hours

2, Calculate and display the parking fee

3, Remove the vehicle using the removeVehicle() method of the ParkingLot object

1.3, Case "Capacity": Display the current car capacity using the getCurrentCapacity() method of the ParkingLot object

1.4, Case "Available": Display the available parking spaces using the getAvailableSpaces() method of the ParkingLot object

Default: Display an error message for invalid input.

4, Case 2: Emergency

1, Prompt the user for their name, emergency type, and location

2. Send an emergency signal using the sendEmergencySignal() method of the EmergencyService object

3. Dispatch an emergency vehicle based on the emergency type using the dispatchEmergencyVehicle() method of the EmergencyService object

4, Clear the road for emergency vehicles using the clearRoad() method of the EmergencyService object

5, Case 3: Pedestrian

1. Prompt the user to activate the crosswalk signal

2. Activate or deactivate the signal based on the user's input using the activateSignal() method of the Crosswalk object

3. Prompt the user to perform an emergency stop

4. Perform an emergency stop if the signal is active using the emergencyStop() method of the Crosswalk object

6, Case 4: Road

1. . Prompt the user for the road capacity
2. Create a Road object with the specified capacity
3. Loop until the user chooses to exit the road module:
4. Display the road options
5. Read and store the user's choice in a variable
6. . Switch based on the user's choice:

Case 4.1: Add Vehicle {Add a vehicle to the road using the addVehicle() method of the Road object}

Case 4.2: Remove Vehicle {Remove a vehicle from the road using the removeVehicle() method of the Road object}

Case 4.3: Check Road Congestion

1. Check if the road isCongested using the isCongested() method of the Road object
2. Display the congestion status

Case 4.4: Exit {Display a farewell message and break out of the loop}

Default: Display an error message for invalid input

7, Case 5: Exit {Display a farewell message and break out of the loop}

Default: Display an error message for invalid input