009

Semaphores

Operating Systems

Lab009 Semaphores

(Smart Parking Lot System)

Objective

Students will implement a simplified smart parking lot management system where multiple cars (threads) attempt to enter and leave a parking lot with limited spaces. The system should ensure that no more cars are parked than available spots, using **semaphores to manage the limited shared resource (parking spots)** and **locks to ensure safe access to shared counters or logs**.

Scenario: Smart Parking Lot System

Imagine a small smart parking lot with only N parking spaces. Cars arrive randomly and try to park. If a spot is available, the gate lets them in. If not, the car must wait. When a car leaves, a spot becomes free, and the next waiting car can be allowed in.
This models real-world parking systems in airports, shopping centers, and smart cities where parking capacity is limited and concurrent access needs coordination.

Assignment Tasks

1. Simulate Car Threads:
- Create a simulation with 10 car threads attempting to access a parking lot with **N** parking spaces.

2. Semaphore for Parking Spots:
- Use a counting semaphore (parking_semaphore) initialized to **N** to represent the number of available parking spots.

3. Thread-Safe Logging:
- Employ a mutex lock (log_mutex) to protect a shared log, ensuring all actions (arrival, parking, leaving) are recorded with timestamps in a thread-safe manner.

4. Car Behavior:
- Each car thread must:
    - **Arrive**: Log its arrival with a timestamp.
    - **Wait**: Wait for an available parking spot.

  - **Park**: Occupy a spot for a random duration (1–5 seconds) and log the parking event.
  - **Leave**: Free the spot and log the departure.

5. Statistics Tracking:
- Use a mutex (stats_mutex) to protect shared counters for total cars parked and total wait time.
- Calculate and report the average waiting time at the end of the simulation.

Expected output:
[Fri Mar 21 13:38:46 2025] Car 0: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 0: Parked successfully (waited 0.00 seconds)
[Fri Mar 21 13:38:46 2025] Car 1: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 3: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 3: Parked successfully (waited 0.00 seconds)
[Fri Mar 21 13:38:46 2025] Car 4: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 6: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 7: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 8: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 9: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 1: Parked successfully (waited 0.00 seconds)
[Fri Mar 21 13:38:46 2025] Car 2: Arrived at parking lot
[Fri Mar 21 13:38:46 2025] Car 5: Arrived at parking lot
[Fri Mar 21 13:38:48 2025] Car 0: Leaving parking lot
[Fri Mar 21 13:38:48 2025] Car 4: Parked successfully (waited 2.00 seconds)
[Fri Mar 21 13:38:49 2025] Car 4: Leaving parking lot
[Fri Mar 21 13:38:49 2025] Car 6: Parked successfully (waited 3.00 seconds)
[Fri Mar 21 13:38:50 2025] Car 3: Leaving parking lot
[Fri Mar 21 13:38:50 2025] Car 1: Leaving parking lot
[Fri Mar 21 13:38:50 2025] Car 7: Parked successfully (waited 4.00 seconds)
[Fri Mar 21 13:38:50 2025] Car 8: Parked successfully (waited 4.00 seconds)
[Fri Mar 21 13:38:51 2025] Car 7: Leaving parking lot
[Fri Mar 21 13:38:51 2025] Car 9: Parked successfully (waited 5.00 seconds)
[Fri Mar 21 13:38:52 2025] Car 6: Leaving parking lot
[Fri Mar 21 13:38:52 2025] Car 2: Parked successfully (waited 6.00 seconds)
[Fri Mar 21 13:38:53 2025] Car 8: Leaving parking lot
[Fri Mar 21 13:38:53 2025] Car 5: Parked successfully (waited 7.00 seconds)
[Fri Mar 21 13:38:53 2025] Car 9: Leaving parking lot
[Fri Mar 21 13:38:54 2025] Car 2: Leaving parking lot
[Fri Mar 21 13:38:58 2025] Car 5: Leaving parking lot
Total cars parked: 10
Average wait time: 3.10 seconds

Extra Points

- Implement a GUI or real-time dashboard