



UNIVERSITI UTARA MALAYSIA

SECOND SEMESTER SESSION 2024/2025 (A242)

STIWK3014 REAL-TIME PROGRAMMING (GROUP A)

ACTIVITIES 10: GROUP DISCUSSION

TRAFFICLIGHTCONTROLLER BY USING REENTRANTLOCK METHOD

PREPARED FOR:

DR. RUZITA BINTI AHMAD

PREPARED BY: GROUP 6

NO.	NAME	MATRIC NO.
1.	NUR AZYYATI BINTI ABU BAKAR	291560
2.	YOGANRAJ A/L LETCHUMANAN	291587
3.	NUR SYAZWANI BINTI MOHD MUZAKIR	291724

THE CODE:

TrafficLightController.java

```
import java.util.concurrent.locks.ReentrantLock;

public class TrafficLightController {
    |
    | enum TrafficLightState { 5 usages
    |     RED, GREEN, YELLOW 3 usages
    | }
    |
    | //state of the traffic light
    | private TrafficLightState currentState; 5 usages
    |
    | //ReentrantLock to protect the shared state
    | private final ReentrantLock lock = new ReentrantLock(); 2 usages
    |
    | //constructor to initialize trafficlight to RED
    | public TrafficLightController() { 1 usage
    |     currentState = TrafficLightState.RED;
    | }
    |
    | //to cycle through the lights
    | public void startSimulation() { 1 usage
    |     while (true) {
    |         lock.lock(); //acquire the lock before modifying shared state
    |         try {
    |             switch (currentState) {
    |                 case RED:
    |                     showRedLight(); //display RED
    |                     currentState = TrafficLightState.GREEN; //next sta
    |                     break;
    |                 case GREEN:
    |                     showGreenLight(); //GREEN
    |                     currentState = TrafficLightState.YELLOW;
    |                     break;
    |                 case YELLOW:
    |                     showYellowLight(); //YELLOW
    |                     currentState = TrafficLightState.RED;
    |                     break;
    |             }
    |         } finally {
    |             lock.unlock(); //always release the lock
    |         }
    |     }
    | }
}
```

```

//Helper method to simulate RED light
private void showRedLight() { 1 usage
    System.out.println("🚦 RED light - STOP");
    sleep(milliseconds: 5000); //wait 5 seconds
}

//Helper method to simulate GREEN light
private void showGreenLight() { 1 usage
    System.out.println("🚦 GREEN light - GO");
    sleep(milliseconds: 4000); //4 seconds
}

//Helper method to simulate YELLOW light
private void showYellowLight() { 1 usage
    System.out.println("🚦 YELLOW light - SLOW DOWN");
    sleep(milliseconds: 2000); //2 seconds
}

//Sleep utility with exception handling
private void sleep(int milliseconds) { 3 usages
    try {
        Thread.sleep(milliseconds); //pause
    } catch (InterruptedException e) {
        System.out.println("Simulation interrupted!");
    }
}

//start the simulation
public static void main(String[] args) {
    TrafficLightController controller = new TrafficLightController();
    controller.startSimulation(); //run
}
}

```

THE OUTPUT:

```

🚦 YELLOW light - SLOW DOWN
🚦 RED light - STOP
🚦 GREEN light - GO
🚦 YELLOW light - SLOW DOWN
🚦 RED light - STOP
🚦 GREEN light - GO
🚦 YELLOW light - SLOW DOWN
🚦 RED light - STOP
🚦 GREEN light - GO
🚦 YELLOW light - SLOW DOWN
🚦 RED light - STOP
🚦 GREEN light - GO
🚦 YELLOW light - SLOW DOWN
🚦 RED light - STOP

```