Kathmandu University
Department of Computer Science and Engineering Dhulikhel,
Kavrepalanchowk

A
Lab Report
on
"Traffic
Simulation using
Queue"

[Code No: COMP202]

Submitted by:

Aayush

Acharya

CS: UNG (2/1)

Roll-no:01

Submitted to:

Dr. Rupak Ghimire

Department of Computer Science and Engineering

Submission Date: 27/12/2025

1)Summary of Work:

This project implements a traffic junction simulator to demonstrate the practical application of linear data structures, specifically Queues, in solving real-world traffic management problems. The simulator models a four-road intersection where vehicles arrive, wait in queues, and are dispatched based on traffic light conditions.

The system supports:

- Normal Conditions: All lanes are served fairly based on occupancy.
- High-Priority Conditions: A designated priority lane (AL2) is served immediately when congestion exceeds a specific threshold

Code:

The link for the GitHub repository is:

https://github.com/Shanks1015/dsa-queue-simulator

2. Problem Statement:

At a junction connecting four major roads (A, B, C, and D), vehicles must be managed efficiently to avoid congestion. Each road consists of three lanes: an incoming lane, a lane controlled by a traffic light, and a free left-turn lane. The system must maintain fairness while dynamically adjusting priority if lane AL2 becomes congested.

3.Data Structure used:

Data Structure	Implementation	Purpose
Queue	Array-based / Struct-based	Store vehicles waiting in each lane [22]
Priority Queue	Condition-based selection	Give priority to AL2 during congestion [23]
Struct	Vehicle, Lane, Traffic Light	Represent simulation entities [24]

4.Algorithm Used:

- Priority Lane Algorithm:
    - Continuously monitor AL2 queue size.
    - If queue size > 10, immediately grant green light to AL2.
    - Continue serving AL2 until queue size < 5.
    - Resume normal scheduling afterward.
- Traffic Light Logic: Only one road can have a green signal at any given time to avoid deadlock.

Time complexity of this algorithm:

- Enqueue: O (1)
- Dequeue: O (1)
- Priority Check: O (1)
- Lane Selection: O(n) where n is the number of lanes.

Output:

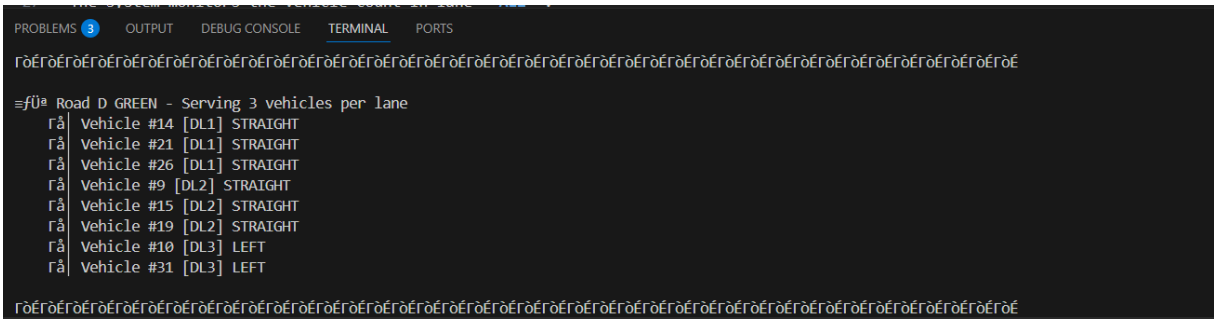

Figure 1: Vehicle direcrtions

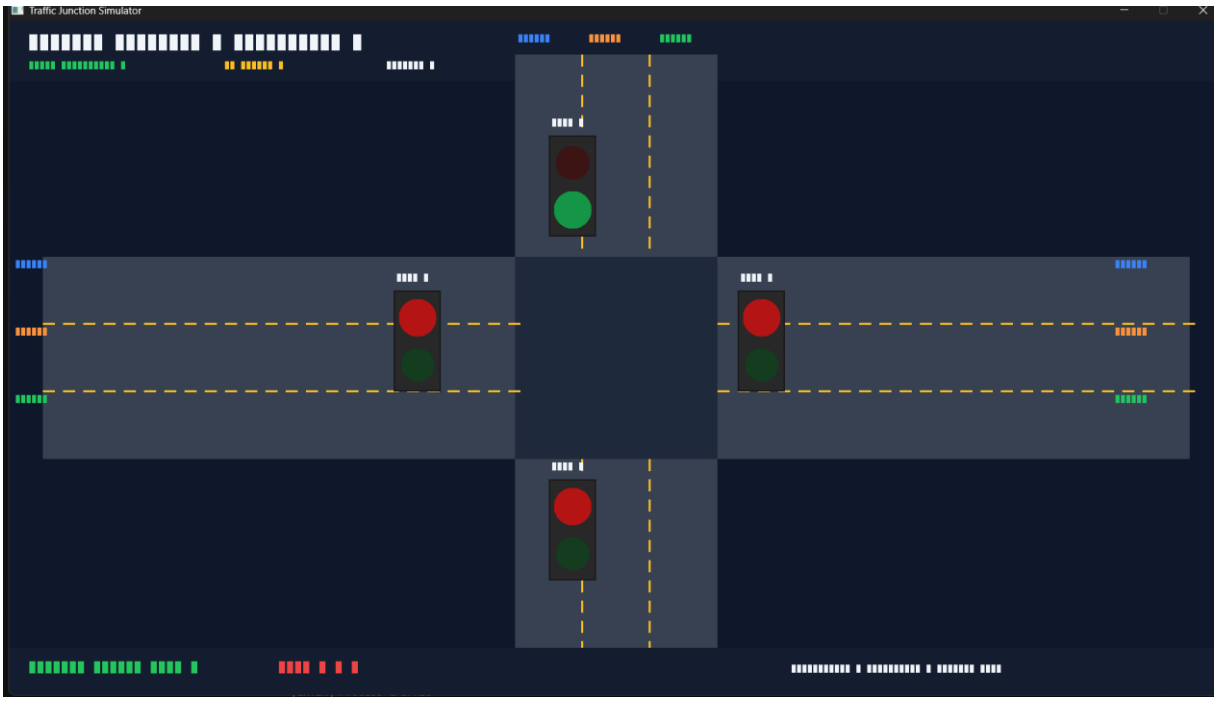

Figure 2: initial window

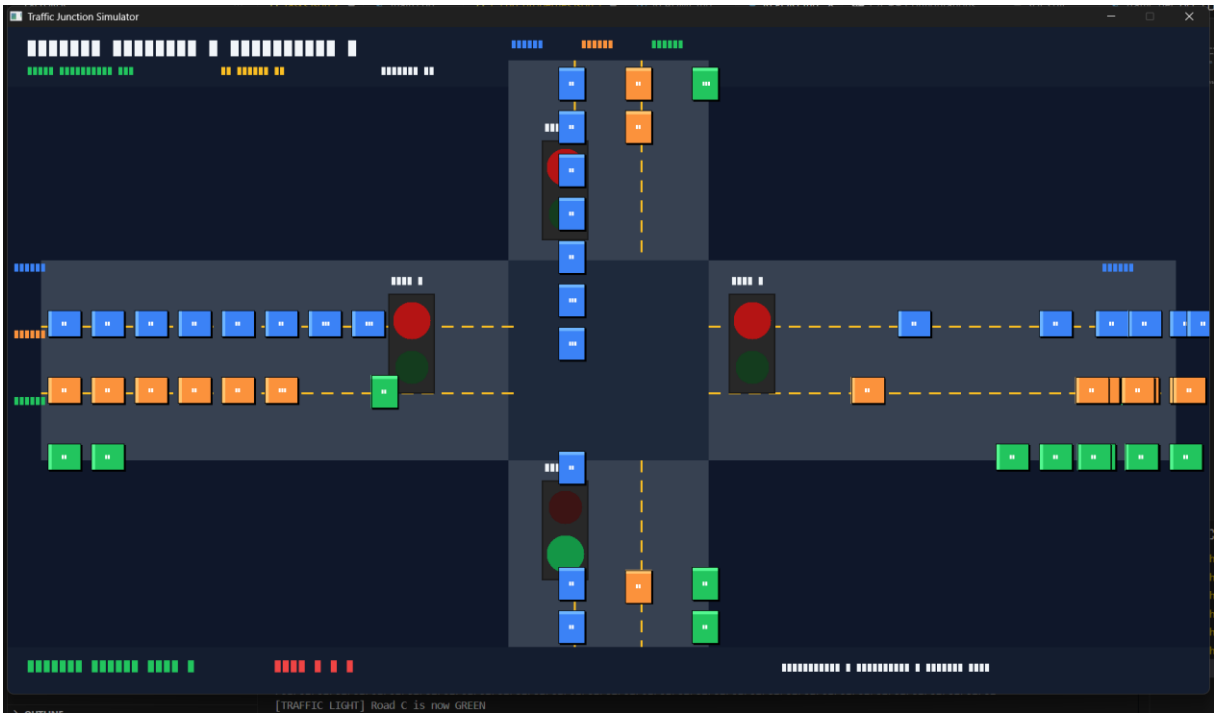

Figure 2: After time

Conclusion:

This project successfully demonstrates how queues can be used to model and solve traffic management problems. By combining data structures with graphical simulation, the system provides both conceptual clarity and practical insight.